N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

CR-152137

(NASA-CR-152137) HYBRID RECEIVER CONCEPTUAL DESIGN AND TEST REPORT Final Report (Motorola, Inc.) 85 p HC A05/MF A01

N81-25279

CSCL 17B

Unclas

G3/32 26035

MOTOROLA INC.

Government Electronics Division

CR 152137

HYBRID RECEIVER CONCEPTUAL DESIGN AND TEST REPORT

Final Report Under
Contract No. NAS 2-9707

31 May 1978

Presented To

Ames Research Center
National Aeronautics and Space Administration

Submitted By

Motorola Government Electronics Division
8201 East McDowell Road
Scottsdale, Arizona 85252

Originators: Stephen W. Klare
Dr. Stephen W. Klare

James J. Crawford
James J. Crawford

MOTOROLA *Government Electronics Division*
8201 E. McDOWELL RD., SCOTTSDALE, ARIZ., 85252

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont)

LIST OF ILLUSTRATIONS

LIST OF TABLES

SECTION 1

1. INTRODUCTION AND SUMMARY

The Motorola design for the Hybrid Receiver uses an acquisition and demodulation scheme tailored for the Jovian environment. The large Doppler of ..is expected during initial acquisition led to the development of a novel acquisition technique. This technique, the Hilbert Acquisition Aid (HAA) has been demonstrated to provide a frequency acquisition characteristic which permits rapid acquisition (≈5 seconds) for low signal-to-noise densities; $(S/N_0) \geq 24$ dB-Hz. (For the demodulation technique, the implementation losses have been shown to be on the order of 0.5 dB from theoretical bit error probability performance. This is a result of its design, where attention has been given to each possible source of performance degradation.)

The hardware implementation of the Hybrid Receiver permits its use in a wide range of applications. It is commandable to permit reconfiguration during separate mission phases as well as programmable to permit its use for varying demodulation techniques, data rates, acquisition algorithms, etc to meet the demands of the communication medium expected for each specific mission. The receiver is also channelized to permit command selection of any of several frequencies at channel separations determined by the particular mission. In addition to its acquisition and demodulation capabilities, spare processing capacity can be used to conduct radio science experiments with the results presented to the telemetry interface as required.

The receiver functional block diagram is shown in Figure 1-1. The current design uses a carrier frequency of 1 GHz with two down conversions prior to the baseband demodulation. The hardware in the RF portion of the receiver, which is derived from the Motorola-designed NASA Standard Transponders and TDRSS (Tracking and Data Relay Satellite System) User Transponder, represents a mature, high-reliability, space-qualified design. The baseband processor, which is derived from the Motorola-designed NASA Standard Command Detector Unit (CDU), provides the receiver with its software control capability and programmable demodulation and acquisition capabilities. As shown in Figure 1-1 the baseband processor accepts I and Q channel inputs from the RF portion of the receiver and controls the frequency channel selection, receiver gain through the Automatic Gain Control (AGC), and frequency and phase of the baseband inputs through the Numerically-Controlled Oscillator (NCO) output. All external interfaces are controlled by the baseband processor with telemetry outputs and command inputs provided as well as lock indication, demodulated data, and clock. Other outputs and inputs can be accommodated as determined in the definition phase of a specific mission.

The important hardware features of this design have been incorporated in a breadboard to verify the associated concepts. The breadboard test results are addressed in later sections of this report; however, the adopted guidelines for the breadboard were:

- The carrier frequency, must be approximately 1 GHz.
- The ability to implement all software algorithms is required.
- Telemetry outputs and radio science processing are not within the scope of this effort.

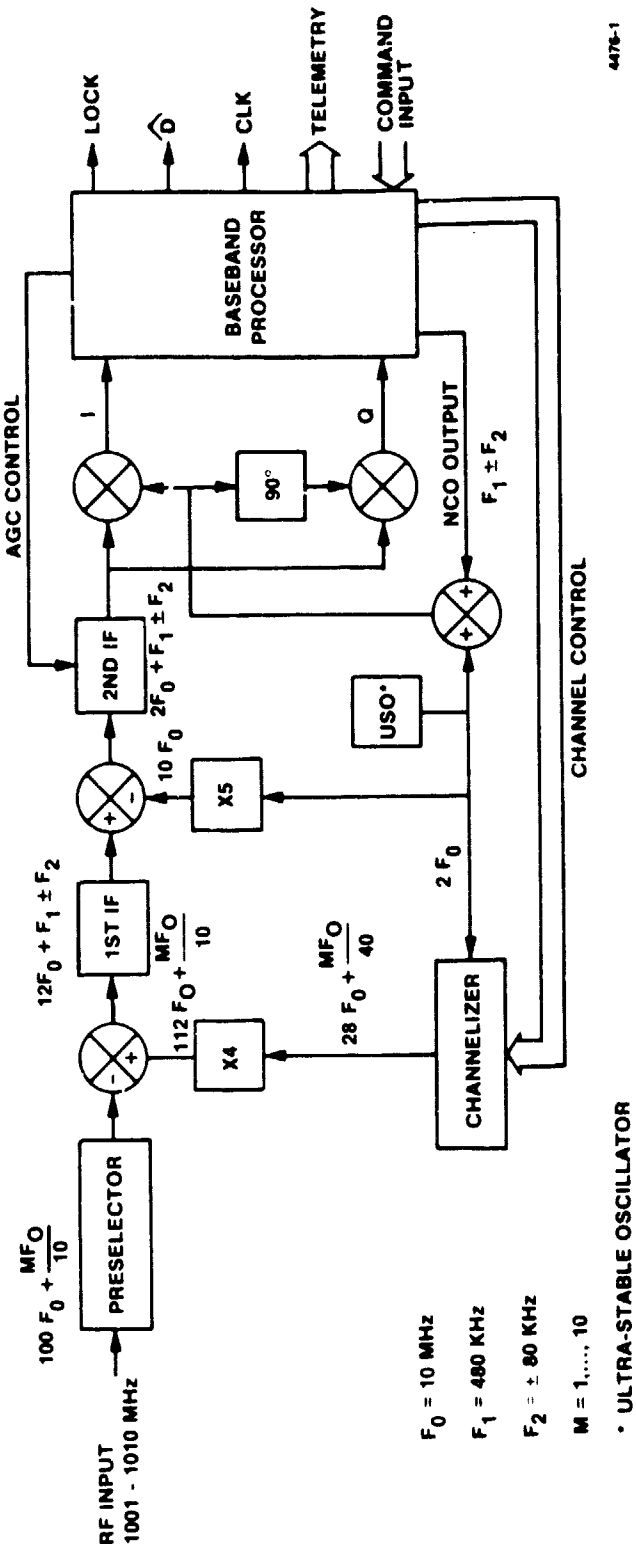


Figure 1-1. Receiver Functional Block Diagram

ORIGINAL PAGE IS
OF POOR QUALITY

Within these guidelines, the HAA and demodulation algorithms were tested with the results presented in Sections 3 and 4.

The circuitry in the Hybrid Receiver will be capable of withstanding radiation doses in excess of 3×10^5 rad (Si) with the RF circuitry comprised entirely of bipolar circuitry. The baseband processor will be implemented in hardened CMOS as well as bipolar circuits. The Motorola-designed LSI circuits used in the baseband processor will be constructed with a hardened, low-power, high-speed silicon-gate CMOS process with proven capability.

The following sections and appendices describe the details of the Hybrid Receiver conceptual design as well as performance analysis and breadboard test results. The functional implementation of the receiver hardware is discussed in Section 2, with the receiver RF design and baseband processor architecture described in detail. The acquisition characteristics of the receiver are addressed in Section 3 with relevant analyses relegated to Appendix C. The demodulation algorithm and the associated bit error probability performance are addressed in Section 4.

Continued development is required in two areas of the Hybrid Receiver design. The first is the acquisition algorithms. The scope of the present efforts did not include the design of a lock detector algorithm. This omission precludes the presentation of acquisition probabilities since without a lock signal the actual acquisition time cannot be determined. The second area requiring additional effort is in the tracking algorithms where the tracking loop bandwidths and responses must be tailored to the actual mission profile. This may require the design of a third order tracking loop. Also, the effects of fading should be determined for the acquisition and tracking algorithm as designed in order that adjustments can be made, if required, to maximize the performance in the expected scintillation environment.

SECTION 2

2. BASIC SYSTEM DESCRIPTION

The Hybrid Receiver is intended to be a highly flexible, channelized design which can acquire and demodulate medium to very low data rate in widely varying communications environments and over a broad range of modulation schemes. This design goal is realized through software control of the important functions of the receiver. Thus, through programs designed specifically for each application, acquisition and demodulation strategies can be tailored for optimal performance. Modulation formats for which algorithms can be developed include: Phase-Shift Keyed (PSK), Differentially Encoded PSK (DEPSK), Differentially Encoded/Detected PSK, (DEDPSK), Quadriphase PSK (QPSK), and Staggered QPSK (SQPSK). For the Jupiter Orbiter Probe (JOP) mission PSK and DEDPSK are under consideration.

2.1 Design Concept

Because of the stability of digital circuitry and the ease with which it can be integrated, the foremost design goal was to digitize the signal as soon in the processing as possible. A secondary goal was to develop an acquisition technique compatible with low signal-to-noise ratios and large frequency offsets. For demodulation the ADC could operate at the IF using coherent sampling. However, for ease of implementation and performance, the acquisition strategy required, a baseband I-Q channel pair. This requirement, coupled with a need to minimize circuitry, led to the decision to demodulate the input signal at baseband. This approach does not compromise the receiver flexibility in any way; however, additional analog circuitry is required.

The basic requirements for the JOP mission are given in Table 2-1. Significant parameters are the minimum ST/N_0 of 4.5 dB and the acquisition probability of 0.995 for a total frequency uncertainty of 160 kHz. Thus, while sufficient attention must be devoted to demodulation of the signal within 1 dB of theory, a much more difficult problem is reliable frequency acquisition.

Table 2-1. Hybrid Receiver Specifications

Minimum ST/N_0	4.5 dB
Minimum Signal Level	−146 dBm
Signal Dynamic Range	30 dB
Noise Dynamic Range	4 dB
Modulation	DEPSK
Data Rate	100 to 1000 bps
Frequency Uncertainty	±80 KHz
Maximum Doppler Rate	50 Hz/s

Table 2-1. Hybrid Receiver Specifications (Cont)

Acquisition Probability, Worst Case in 50s	0.995
False Acquisition Probability, Worst Case in 50s	<0.0001
Maximum Frequency Tracking Error for 1s Integration, at Maximum Signal Strength.	1 Hz rms
Detection	DEDPSK
Bit Error Performance (0.1 to 5 × 10^{-1}),	≤1 dB of theory
Signal Power Resolution	±0.1 dB

Motorola has proposed a new acquisition strategy, the HAA, to meet this requirement.

The receiver block diagram shown in Figure 1-1 has been divided into an RF section and a baseband processor to provide more detail. The RF section shows the interfaces between the baseband and RF processor section and the input signal (Figure 2-1). Both the first and second IF strips are at fixed frequencies with the channel offsets extracted by the channelizer under software control of the digital processor. The AGC is also controlled through software algorithms as is the offset frequency estimate ($F_1 \pm F_2$). Thus, except for the channelizer, which requires a phase-locked loop design, the RF hardware is static. This is the essence of the design concept, where all of the important features of the receiver are programmable. The breadboard design, which was used to demonstrate the feasibility of this concept, is shown in Figure 2-2.

The baseband processor is shown in Figure 2-3. This processor makes use of the Motorola custom microprocessor chips designed for the NASA Standard CDU. The CDU, minus its analog interfaces, has been used as the digital processor in this design with the primary external interfaces compromising one of its functions. While the software algorithms must respond to the command inputs, simultaneous acquisition and demodulation are achieved through separate algorithms controlling carrier acquisition and tracking, data detection, AGC adjustments, and input channel selection. The input frequency offset and phase are controlled through the NCO interface. Acquisition and tracking inputs are obtained through two separate I-Q channel pairs. One pair is wideband for acquisition of large frequency uncertainties while the other is maintained at a bandwidth greater than or equal to three times the selected data rate. Digitization of these signals is performed as close to the final mixers as is practical to avoid the problems of drift and dc offsets associated with the analog circuits.

The physical characteristics of the Motorola design are shown in Table 2-2. Through the use of LSI and low power RF designs, the size, power and weight have been minimized. With a power consumption of 7.3 watts, a weight of 2050 grams and volume of 2100 cubic centimeters the Motorola Hybrid Receiver represents a refined state-of-the-art digital receiver with a flexibility to meet the requirements of widely varied missions through software changes.

ORIGINAL PAGE IS
OF POOR QUALITY

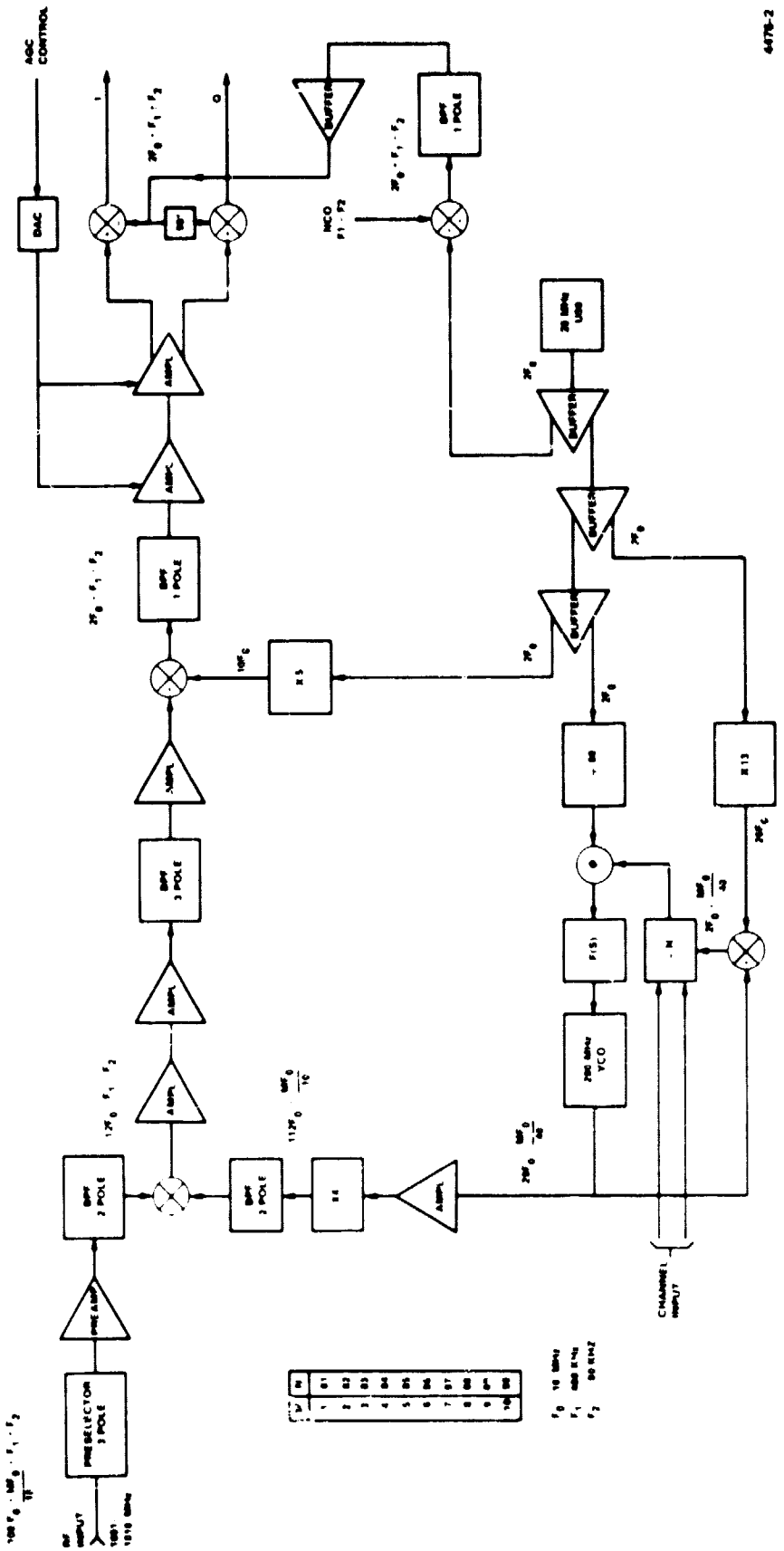


Figure 2-1. Hybrid Receiver RF Design

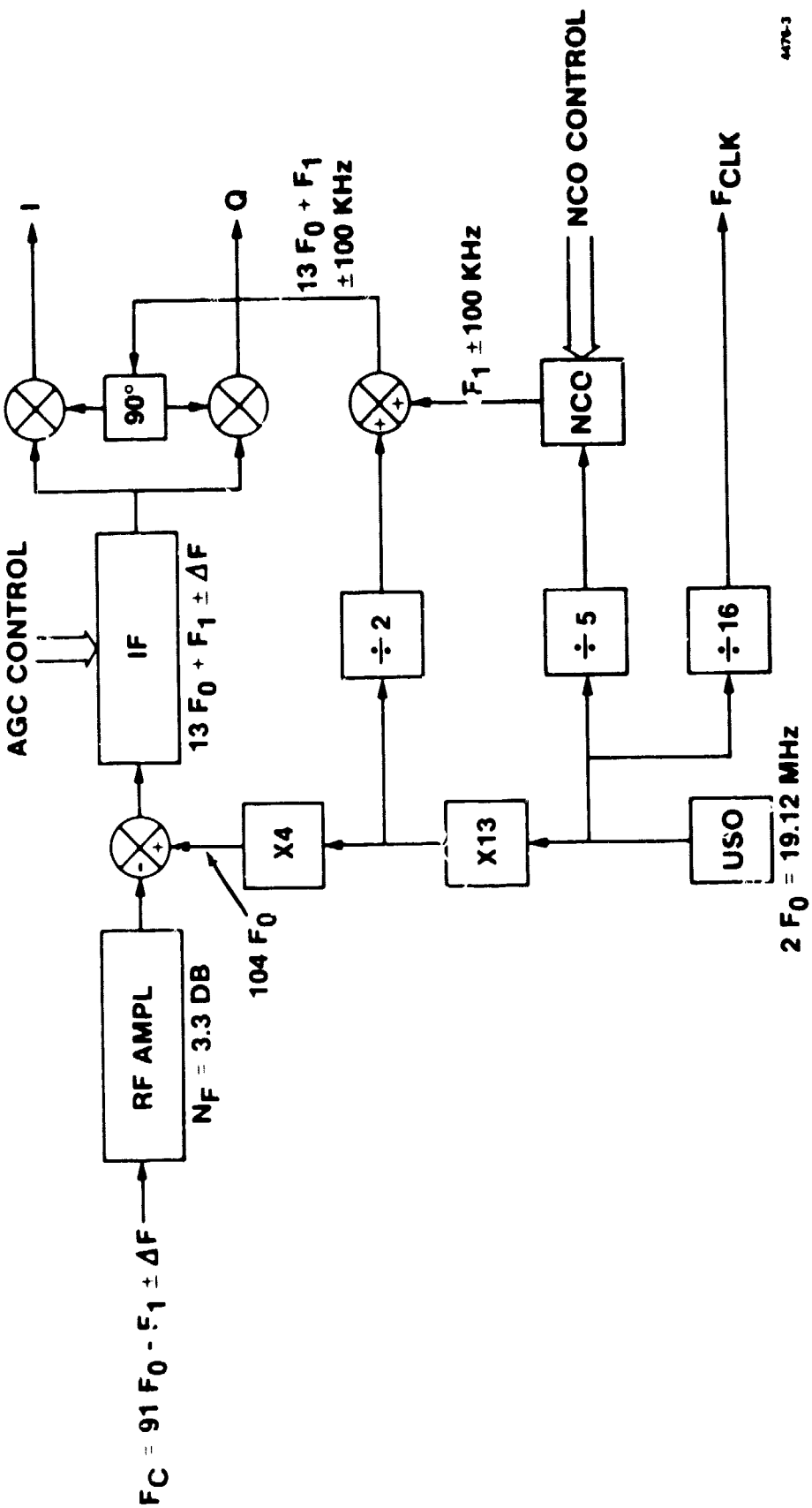


Figure 2-2. Hybrid Receiver RF Breadboard

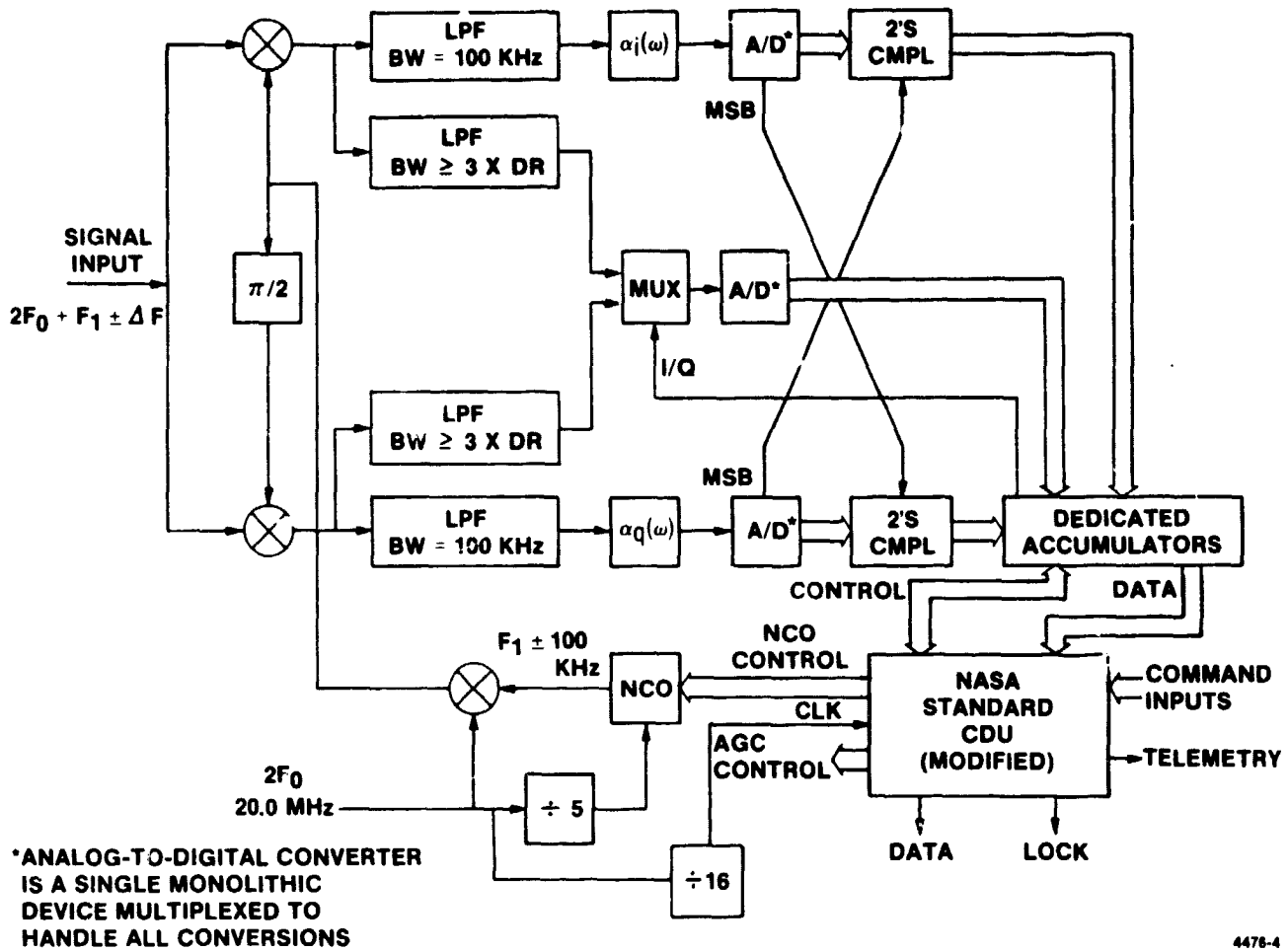


Figure 2-3. Hybrid Receiver Baseband Processor

Table 2-2. Hybrid Receiver Physical Characteristics

	Size (cm)	Volume (cm^3)	Weight (gm)	Power (W)
Baseband	14 × 20 × 3	840	250	3.0
Power Converter	14 × 6 × 1.5	126	600	1.8
RF	14 × 20 × 3	840	1200	2.5
Spare Volume	14 × 14 × 1.5	294		
Hybrid Receiver	14 × 20 × 7.5	2100	2050	7.3

2.1.1 ACQUISITION

In order to achieve the best performance, a wideband frequency acquisition loop has been designed to give reliable acquisition performance for the low signal-to-noise and large uncertainties to be encountered during the JOP mission. This technique avoids the complexities of sequential detection with its thresholds, dismiss times, cell widths, etc and avoids the problems associated with discriminator-aided acquisition. The HAA, described in detail in Section 3, has been designed to give a reliable and rapid acquisition characteristic with none of the difficulties of the previously mentioned schemes and with superior performance. Its salient feature, the Hilbert transform approximation, provides a unique acquisition aid.

2.1.2 CARRIER TRACKING AND DATA DETECTION

Carrier tracking is effected through software algorithms deriving frequency and phase estimates which are then input to the NCO. This approach allows very stable carrier tracking as well as accurate telemetry information as to the frequency of the input signal. A Maximum A Posterioi (MAP) tracking scheme, which reduces the carrier tracking error to the minimum achievable for a given loop bandwidth, is described in Section 4. Superior performance is achieved through a data-aided approach. Thus, the two algorithms, MAP carrier tracking and bit synchronization are designed in tandem to enhance the performance of both algorithms through synergic interaction.

2.1.3 AUTOMATIC GAIN CONTROL

The design of the AGC algorithms has yet to be completed for the JOP application. This function is totally software controlled in order that the bandwidths and coherence or noncoherence of the strategy can be programmable to achieve the best performance for the particular mission. The bandwidth of the AGC loop is envisioned to be on the order of several Hz, which would permit it to track the amplitude spectrum to be experienced in the scintillation environment of Jupiter. Currently a coherent algorithm, which will be similar to that used in the CDU is envisioned; however, the acquisition difficulties may require a noncoherent AGC (NAGC) for optimum acquisition performance. This is easily accomplished in the software and will be designed when the final mission parameters become available. A 40-dB dynamic range is envisioned for the AGC with 10-bit resolution, or ±0.04 dB resolution.

2.1.4 LOCK DETECTOR

The lock detector, also a software algorithm, will be designed at a later time since its performance will not enhance the demonstration of feasibility of the HAA or the tracking algorithms. Its design will be based on the design of the lock detector of the CDU with its high degree of analysis and experimental verification.

2.2 Digital Subsystem-Hardware Implementation

The digital subsystem of the Hybrid Receiver is implemented using a custom CMOS microprocessor. A microprocessor approach was chosen mainly because of the ease with which the system algorithms can be modified in the software. This approach also has the advantages of lower power requirements, less weight and smaller size than a hardwired MSI configuration.

The microprocessor used in the digital subsystem is structured as shown in Figure 2-4. In order to optimize data handling operations the system is configured using a 10-bit address bus, a 16-bit instruction bus and a 16-bit data bus. By keeping these three buses separate the system is capable of performing data fetches or executing instructions in a single system clock cycle.

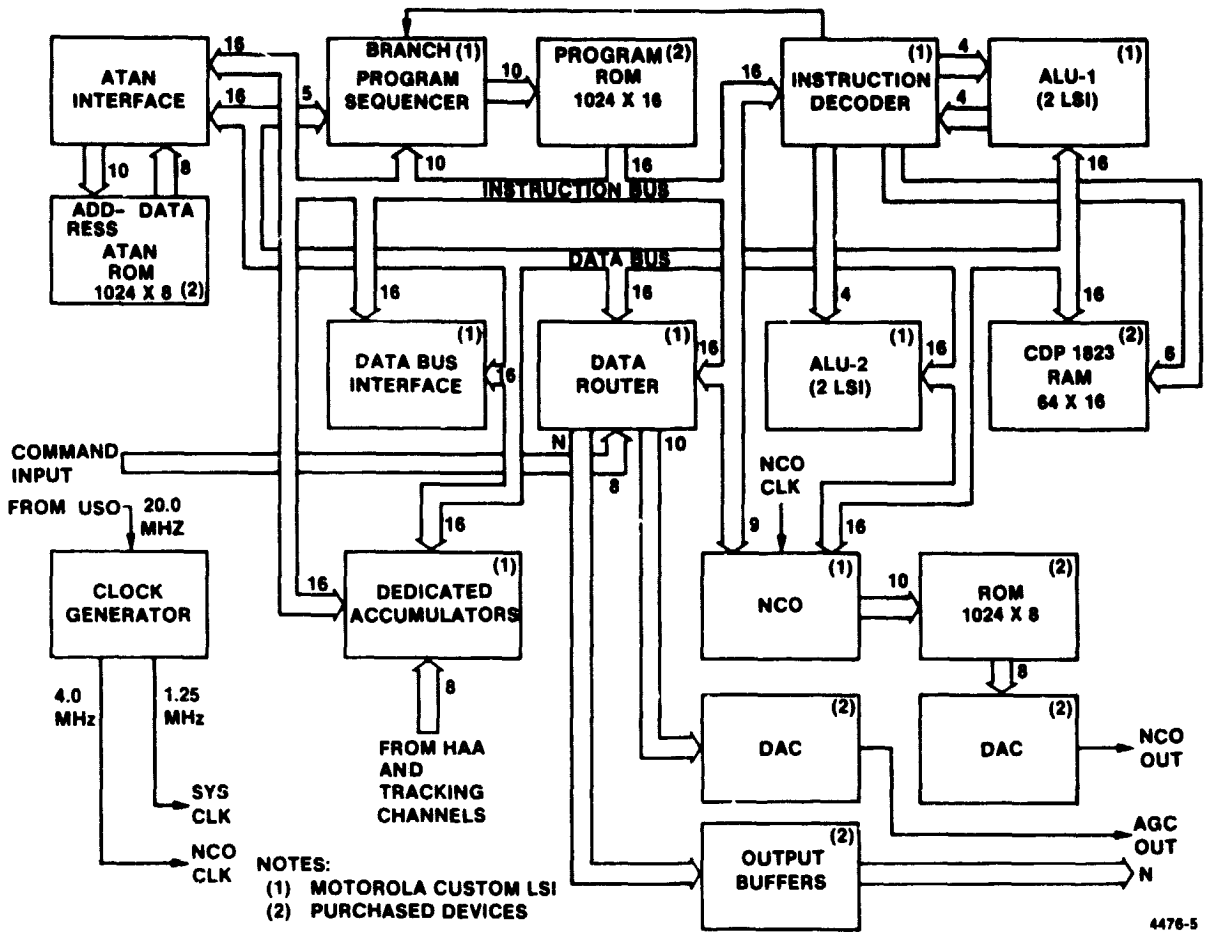


Figure 2-4. Hybrid Receiver Digital Subsystem

Operation of the processor is controlled by instructions stored in the program Read-Only Memory (ROM). Instructions are read from the program ROM under control of the 10-bit address on the address bus and are transferred, via the instruction bus, to the other devices for execution. Transfer of data between devices is accomplished by means of a bidirectional, 3-state data bus.

A brief summary of the functions of each device in the processor is given in Table 2-3. Appendix B contains a detailed description of the microprocessor architecture and instruction set; Table 2-4 summarizes the size and power requirements for the Motorola custom Large-Scale Integrated (LSI) circuits.

Table 2-3. Processor Device Functions

PROGRAM SEQUENCER

- Controls sequencing of the stored ROM program
- Implements program branches

ORIGINAL PAGE IS
OF POOR QUALITY

Table 2-3. Processor Device Functions (Cont)

INSTRUCTION DECODER

- Controls the read/write instructions to the RAM
- Separates ALU instructions from the instruction word out of the ROM
- Decodes branch instructions to be implemented in the program sequencer

DATA BUS INTERFACE

- Controls the read/write instructions to other devices
- Controls the selection of condition flags, constants, data rate code or ROM words to be placed on the data bus
- Contains the shift count register for control of Arithmetic Logic Unit (ALU) shift operations

ALU

- Performs two's-complement arithmetic operations required by the program (add, subtract, negate, absolute value, increment and arithmetic shift right).
- Generates condition code flags used for program branches (carry, overflow, negative and zero).

DATA ROUTER

- Interfaces the command input to the data bus
- Stores and outputs the AGC word
- Controls the timing of output signals

DEDICATED ACCUMULATORS

- Accumulates samples from A/D converter
- Generates A/D converter and sample/hold timing

NUMERICALLY CONTROLLED OSCILLATOR

- Generates Local Oscillator (LO) signal for second IF
- Allows frequency and phase control by a digital word

RANDOM ACCESS MEMORY (RAM)

- Contains storage for program variables
- 64 words × 16 bits

PROGRAM ROM

- Contains the stored program for control of the processor
- 1024 words × 16 bits

2.2.1 PROGRAM SEQUENCER

The purpose of the program sequencer is to control the order of execution of instructions from the program ROM and to implement program branches. The program sequencer (Figure 2-5) consists of a presettable 10-bit synchronous counter, a 2:1 multiplexer, a 10-bit full adder and a 5-bit register. In normal operation the program counter is incremented every system clock cycle and instructions are read from the program ROM in sequence. In

Table 2-4. Motorola Custom CMOS Size and Power

Device	Chip Dimensions (Mils)	Number of Transistors	Power Consumption (mW)	
			Metal Gate f = 1.024 MHz V_{DD} = 13.4V	Si-Gate (est) f = 1.25 MHz V_{DD} = 5V
Program Sequencer	204 × 212	1058	167	50
ALU (each)	219 × 219	1156	51	15
Instruction Decoder	220 × 195	998	176	53
Data Bus Interface	244 × 239	1279	119	36
Data Router	229 × 229	1522	63	19
Dedicated Accumulator	232 × 232 (est)	1200 - 1500 (est)	—	50
Numerically Controlled Oscillator	255 × 255 (est)	1200 - 1500 (est)	—	100

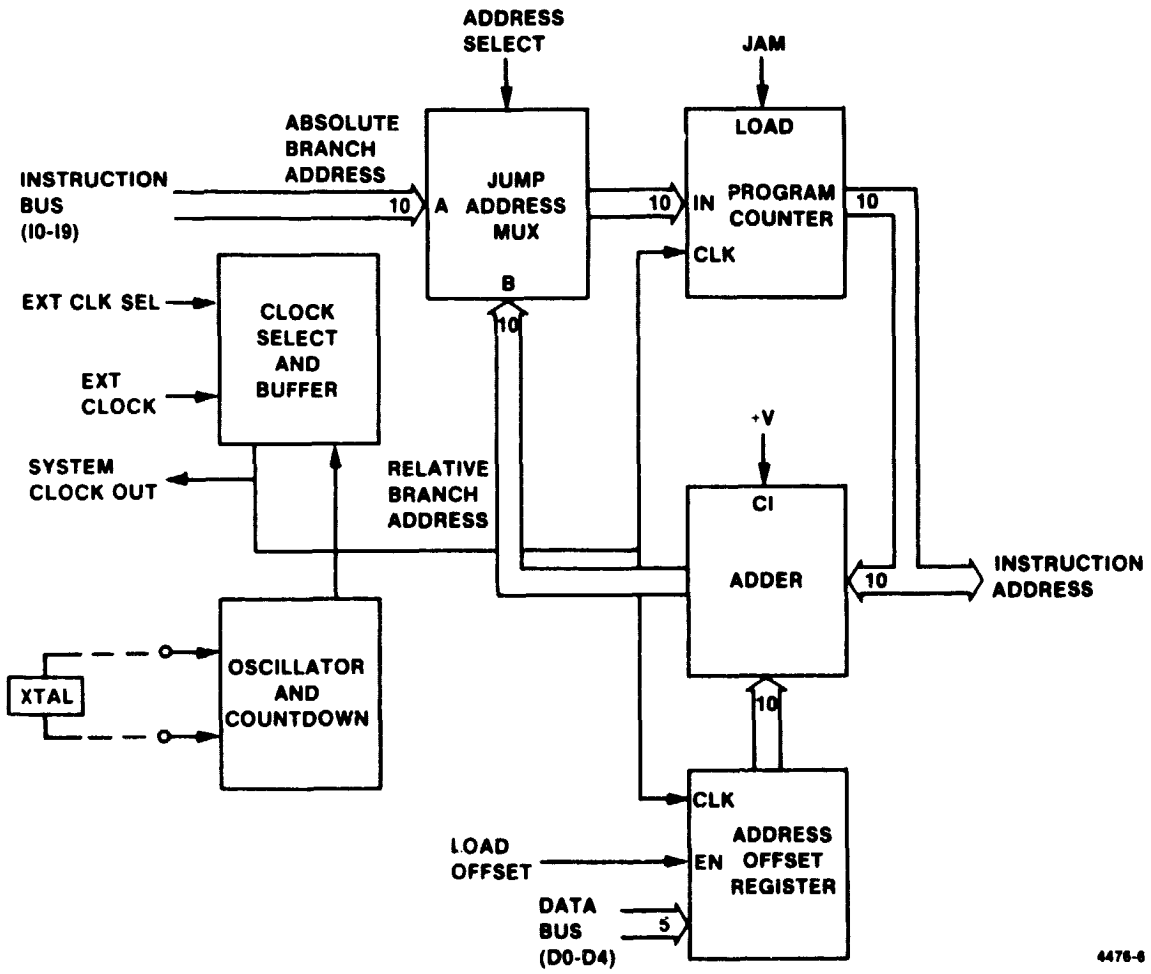


Figure 2-5. Program Sequencer

order to execute branch instructions, however, it is necessary to force the address bus to some arbitrary state and to continue program execution from that point. A program branch is performed by synchronously loading the program counter with the address on the output of the jump address multiplexer. The jump address multiplexer selects the proper address depending on the type of branch to be performed. In the case of an absolute branch, bits 0 through 9 of the instruction bus are selected as the next instruction address. For a relative branch the number in the address offset register is added to the current address plus one and the result becomes the next address. The counter preset and address selection are controlled by the JAM and ADDRESS SELECT signals, respectively, from the instruction decoder.

2.2.2 INSTRUCTION DECODER

The instruction decoder serves three main functions in the system: generation of the 4-bit ALU instruction words, Random-Access Memory (RAM) access control, and program branch decoding. The ALU instructions are generated by the ALU instruction decoder. The ALU instruction decoder separates ALU instruction words from the word on the instruction bus and sends them to the proper ALU for execution. The actual decoding of the 4-bit instruction word is performed by each ALU. Instructions that involve the RAM are executed under control of the RAM access decoder. The RAM access decoder separates the RAM address from the 16-bit instruction word and generates the chip select and memory read signals required for operation of the RAM. The branch decoder controls the execution of the program branch instructions. Program branches can be made conditionally on the state of any of the eight program flags, the four condition-code flags (C, V, N, Z), or the three externally controlled user flags. The branch decoder determines whether or not the state of the selected flag meets the condition required for a branch to take place; if so it generates a JAM signal to the program sequencer. The branch decoder also selects the correct branch address depending on whether an absolute or a relative branch is to be performed. The block diagram for the instruction decoder is shown in Figure 2-6.

2.2.3 DATA BUS INTERFACE

The Data Bus Interface (DBI) shown in Figure 2-7 provides the means for interfacing the system data bus to various external data sources. Program constants contained in the program ROM, the 4-bit data rate code, and the condition-code and input flags can all be placed on the data bus through the DBI. When program constants are read from the program ROM it is necessary to disable the instruction decoders on each of the chips to prevent them from responding to the constant as though it were an instruction. The IREC signal, generated on the DBI, serves this function. Also implemented on this chip are the two 5-bit shift counters required for ALU shift operations. These counters control the length of shifts performed by the ALU chips.

2.2.4 DATA ROUTER

The data router chip controls the interfaces between the digital subsystem and external subsystems. Circuitry for interfacing the command input and the AGC to the processor is implemented on this chip. The required output signals from the digital subsystem, such as detected data and status indicators, are generated on the data router. The data router also provides a 16-bit serial output word at a rate of one-eighth of the system clock for telemetry use (the SNR signals shown in Figure 2-8). The control signals for the data router are supplied by the DBI chip.

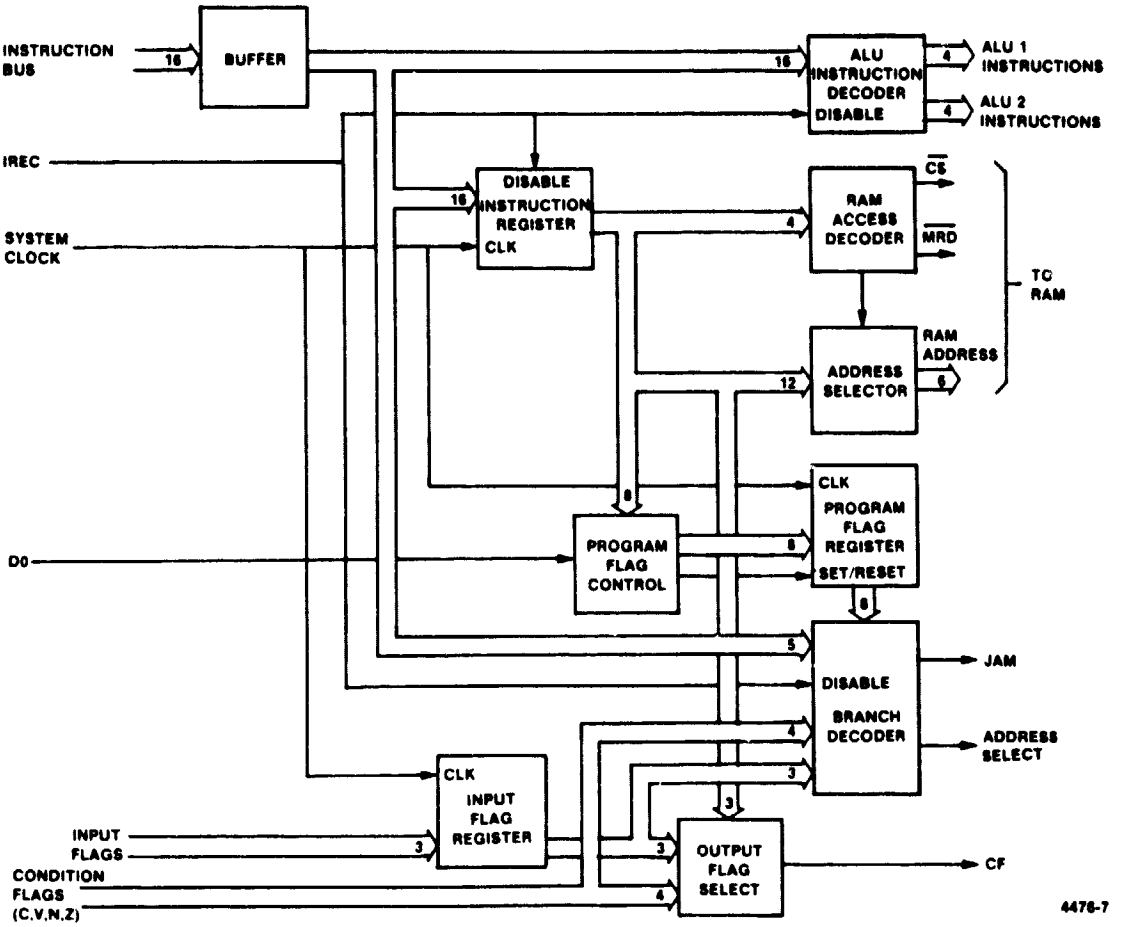


Figure 2-6. Instruction Decoder

2.2.5 ALU

The ALU is a 16-bit arithmetic unit (organized as two 8-bit slices) that performs 2's-complement arithmetic operations. This unit is capable of performing 13 operations, which are coded on a 4-bit instruction word (detailed in paragraph D.1.2.3 of Appendix D). Operand data is stored in the A and B registers with results stored in the B register. The ALU provides four condition code flags (zero, overflow, carry and negative) that are used to perform conditional branches. All ALU instructions are executed at the system clock rate. A 16-bit add operation is executed in four clock cycles as follows: (1) load A register, (2) load B register, (3) add, and (4) read B register. A block diagram of an 8-bit slice ALU is shown in Figure 2-9.

2.2.6 DEDICATED ACCUMULATORS

The dedicated accumulators perform accumulations on the samples from the HAA (acquisition) and tracking channels. Samples from the A/D converter are loaded into registers and added to the appropriate accumulator. This chip contains six 16-bit accumulators that can be read or written into by the processor. The A/D converter and sample/hold control signals are generated on this chip. The block diagram for the dedicated accumulators is shown in Figure 2-10.

ORIGINAL PAGE IS
OF POOR QUALITY

2.2.7 NUMERICALLY CONTROLLED OSCILLATOR

The NCO generates a sinusoidal signal of which the frequency and phase are controlled by a digital word from the processor. As shown in Figure 2-11, the NCO has a 24-bit input register (loaded as two 16-bit bytes), a 24-bit full adder and a 24-bit storage register. The binary number in the input register controls the NCO frequency according to the formula:

$$f_{out} = \frac{\text{Control Word}}{2^{24}} \times f_{clk} \qquad (2\text{-}1)$$

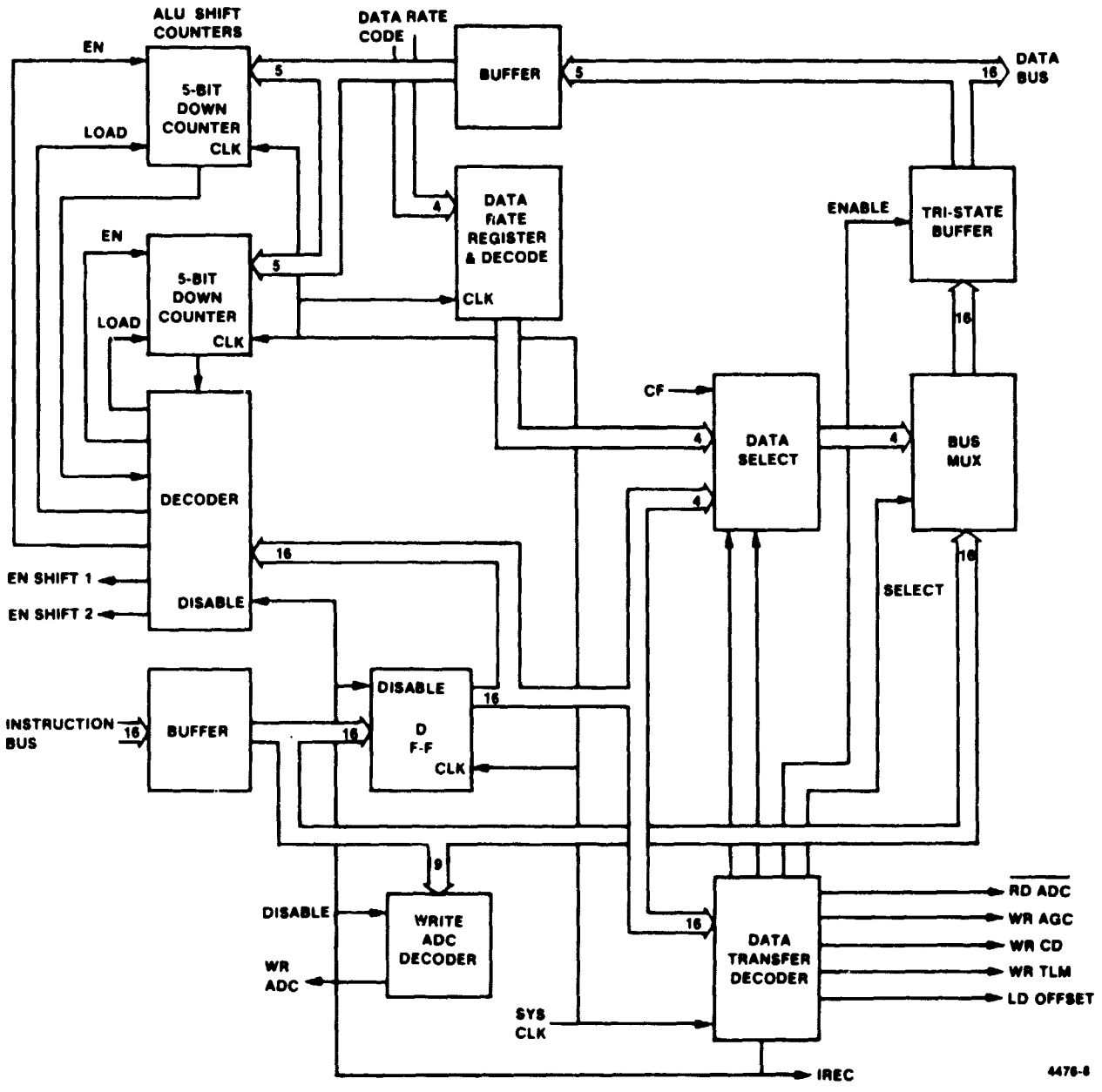


Figure 2-7. Data Bus Interface

where f_c is the NCO clock frequency, f_{nco} is the output frequency and the control word is the binary number in the input register. A read-only memory (external to the NCO) maps the frequency word to a magnitude by means of a D/A converter. Phase control of the output is provided through the phase register. By changing the number in the phase register, the phase of the output sinusoid can be instantaneously shifted by up to ±180 degrees. The incremental phase shift is given by:

$$\Delta\phi = (\text{Phase Word}) \times \frac{360 \text{ deg}}{1024} \qquad (2\text{-}2)$$

where the phase word is in the range from −1024 to +1023.

The output of the D/A converter is amplified and filtered and fed to the RF section of the receiver.

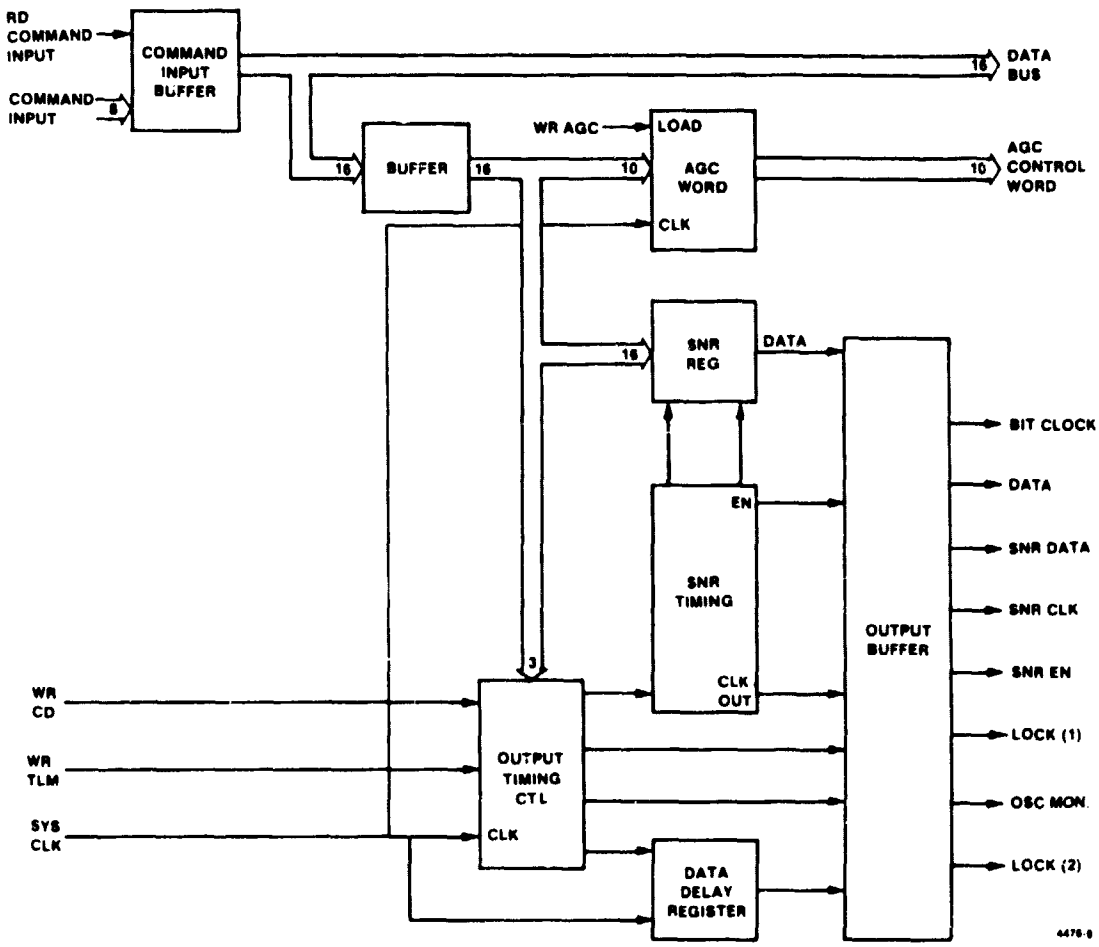


Figure 2-8. Data Router

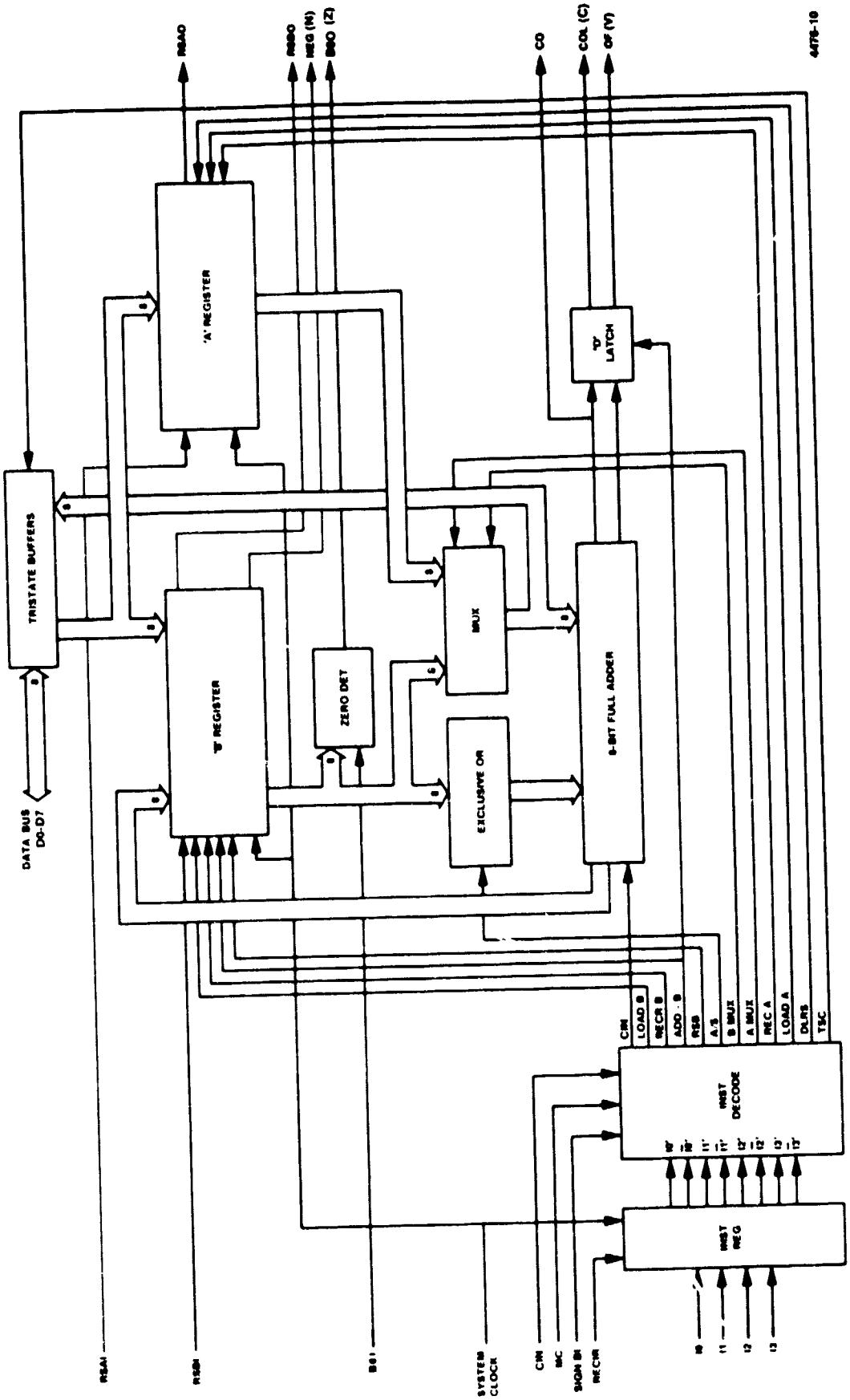


Figure 2-9. 8-Bit Slice ALU Block Diagram

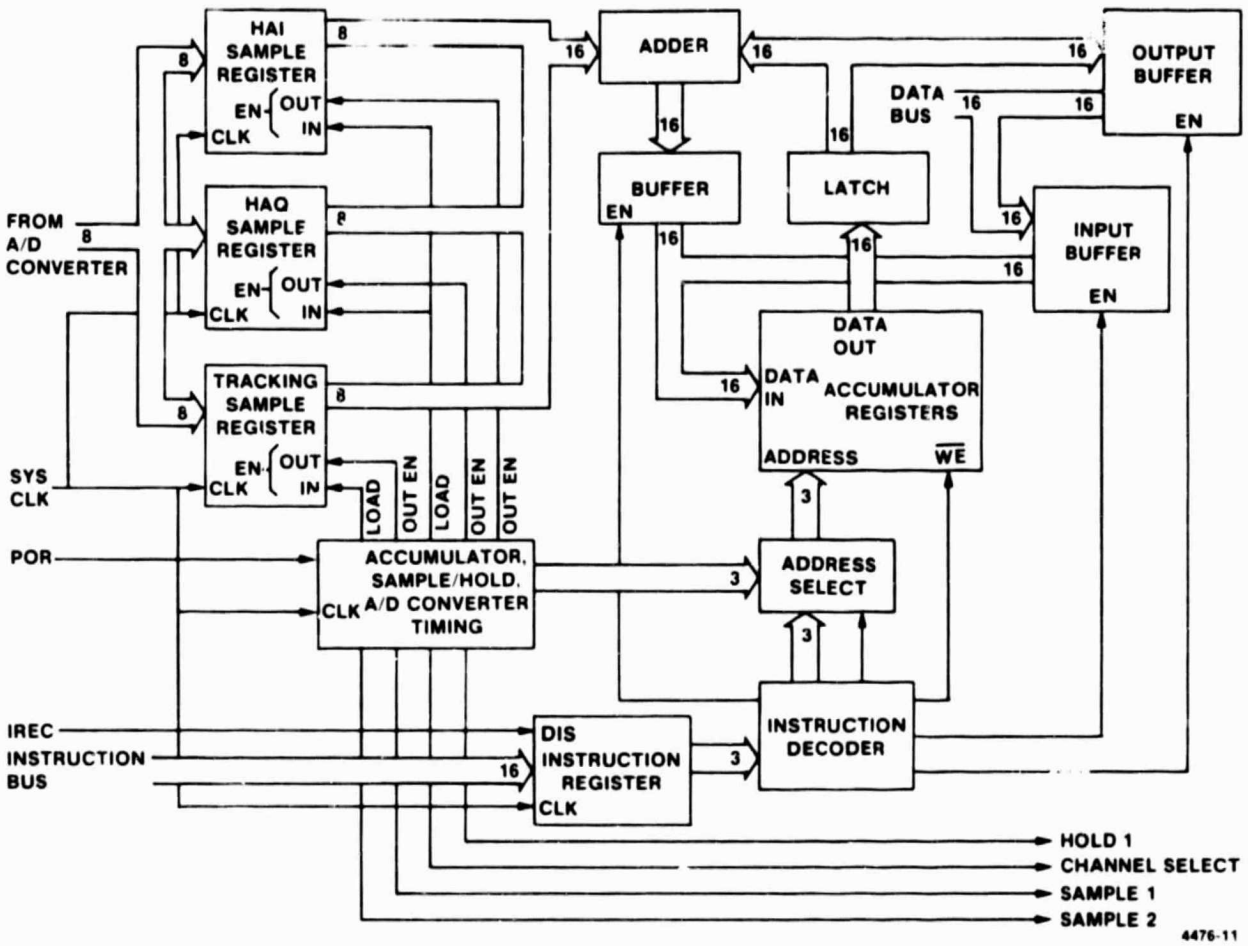


Figure 2-10. Dedicated Accumulators

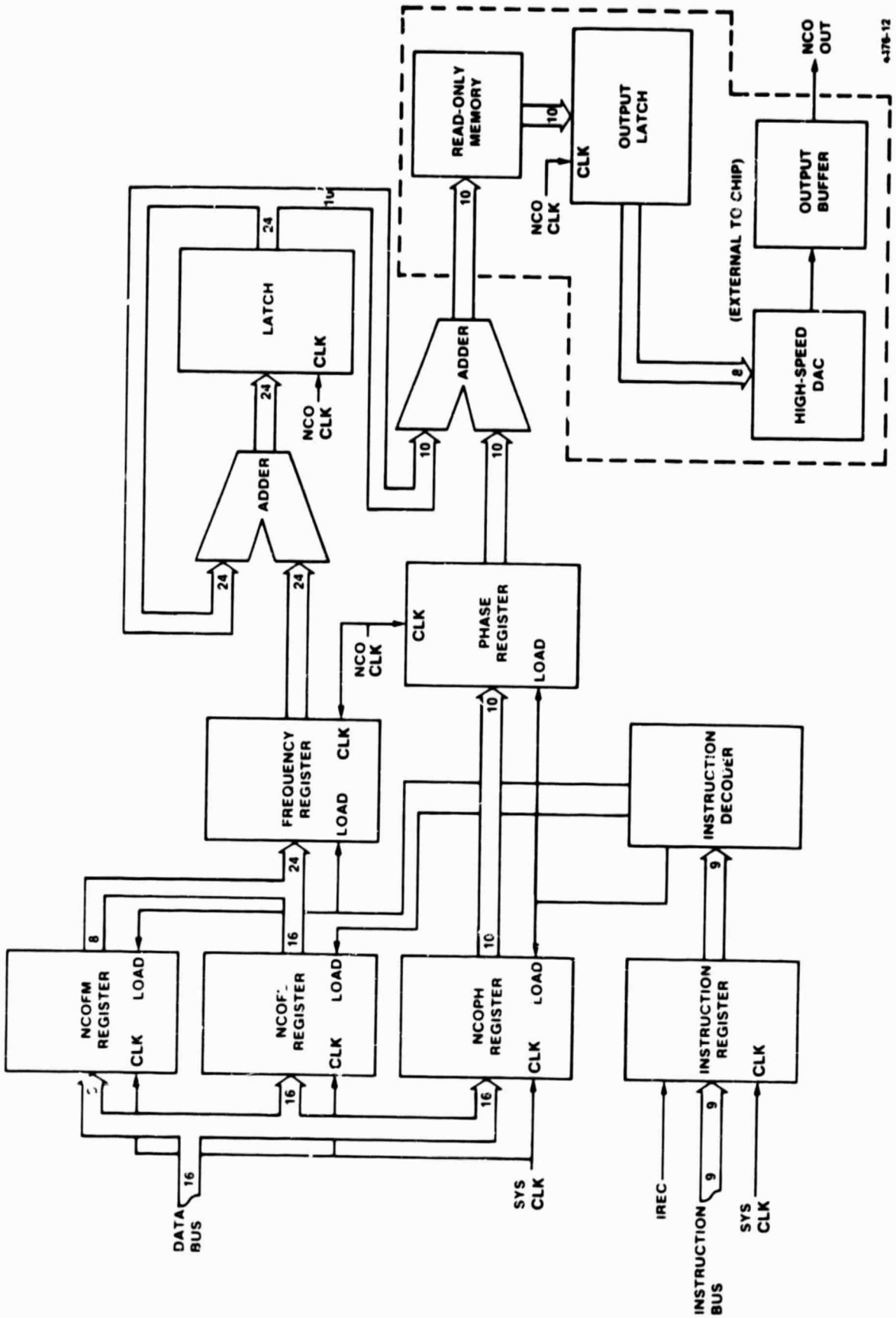


Figure 2-11. Numerically Controlled Oscillator

ORIGINAL PAGE IS
OF POOR QUALITY

SECTION 3

3. HILBERT ACQUISITION AID

The development of the HAA was necessitated by what Motorola considered undesirable properties of the sequential-detection-based acquisition strategies. Whereas sequential detection requires a somewhat complex set of thresholds and dismiss times, the HAA requires only a loop noise-bandwidth setting established by the particular design point selected by the mission. The HAA has a gradual degradation in acquisition time as the input SNR drops below the design point and acquires dramatically faster as the SNR is improved. A functional block diagram of the HAA is shown in Figure 3-1.

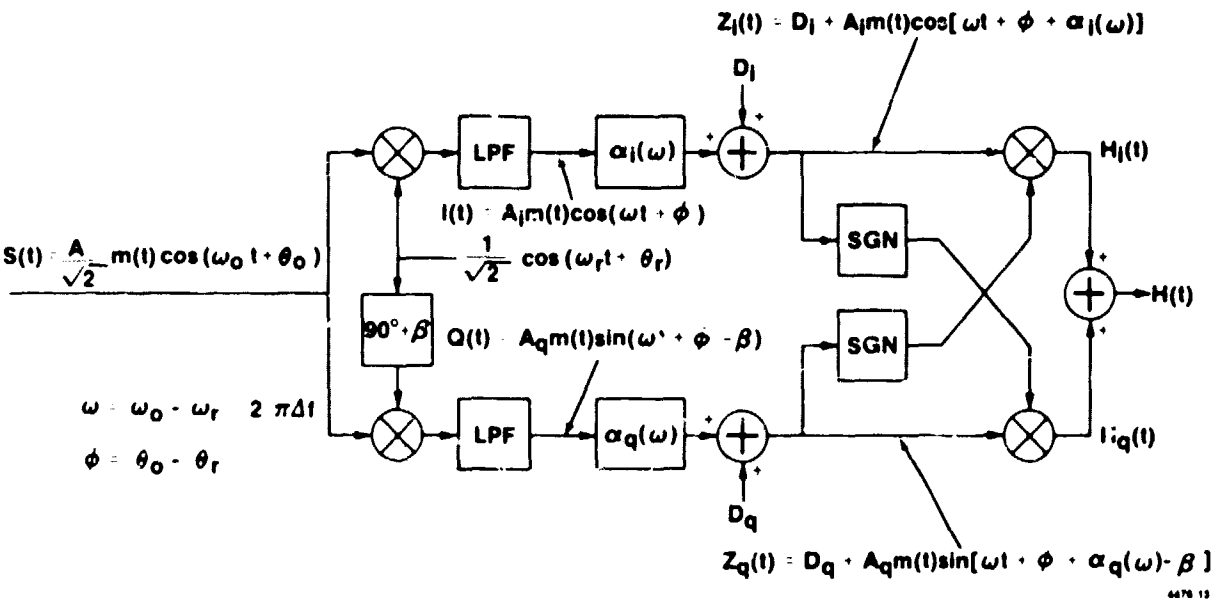


Figure 3-1. Hilbert Acquisition Aid

3.1 HAA Analysis

A noiseless analysis of this scheme was undertaken (Appendix C) to determine the effects of dc offsets, quadrature error, amplitude imbalance, and limiters in the cross-terms.

For the case where the SGN operators were ignored (i.e., no limiting) the resulting control signal [<H(t)>] is derived in (C-9) which is repeated here for reference.

$$\langle H(t) \rangle = 2\, D_I D_Q - A_I A_Q \sin(\psi + \beta) \tag{3-1}$$

where

D_I and D_Q = I and Q channel dc offsets

A_I and A_Q = I and Q channel signal amplitudes

β = Quadrature error

ψ = $\alpha_I(f) - \alpha_Q(f)$

In arriving at this result it was assumed that the control signal was averaged for a period of time which is large with respect to the offset frequency, $(\Delta f)^{-1}$. Seen in (3-1) is that the dc offsets (D_I and D_Q) are not affected by data modulation or reference phase and must be controlled to avoid significant error. However, if an adequate transition density exists in the data modulation, or if there are periodic software-controlled phase inversions, capacitive coupling is permitted, allowing the offsets to be reduced to only two elements in the actual implementation - one buffer amplifier and an Analog-to-Digital Converter (ADC). Again, from (3-1) it is seen that quadrature errors (β), if not controlled adequately, can produce significant degradation. With reasonable care β can be reduced to negligible proportions and then only the I-Q phase difference becomes important.

An all-pass network has been designed which permits the I-Q phase difference (ψ) to be controlled over a wide range of frequencies with a characteristic which approximates a Hilbert transform from which this technique derives its name. Typical characteristics of $\psi(\omega)$ are shown in Appendix C, Figures C-2 and C-5. The purpose of the HAA is to resolve the frequency within a sufficiently narrow band in order that a coherent phase-tracking algorithm can acquire it. Thus, a characteristic as shown in Figure C-3 would be used for loop bandwidths (B_L) on the order of 100 Hz and the characteristic shown in Figure C-6 would apply for $B_L \approx 10$ Hz. For the JCP mission B_L must be approximately 5 to 20 Hz. Therefore, the second characteristic has been selected.

Because the Hybrid Receiver design is primarily digital with the maximum use of software control, the use of limiters permits the digitization of the I and Q acquisition channels prior to the multiplication with only sign inversions for multipliers. Thus the requirement for an 8 bit x 8 bit multiplier and its associated high power consumption is removed. Therefore, the technique with the limiters has become the primary alternative. The analysis of this scheme is undertaken in Appendix C with the same conditions as were previously assumed for no limiters. The resulting control signal - $H(t)$ - is given in (C-28) and (C-29) and is repeated here for reference.

$$\langle H(t) \rangle = -\frac{2}{\pi}\left[D_I \phi_{-1}(2p-1) + D_Q \phi(2p-1) + A\cos\phi_{-1}\sin(\psi+\beta) + A_Q\cos\phi_1\sin(\psi+\beta)\right] \tag{3-2}$$

$$|D_I| < |A_I|$$
$$|D_Q| < |A_Q|$$

$$= D\,SGN(D_Q) + D_Q SGN(D) \tag{3-3}$$

$$|D_I| \geq |A_I|$$
$$|D_Q| \geq |A_Q|$$

where

$$\phi_i = \sin^{-1}\left(\frac{D_i}{A_i}\right);\ \phi_q = \sin^{-1}\left(\frac{D_q}{A_q}\right);\ \psi = \alpha_i(\omega) - \alpha_q(\omega) \qquad (3\text{-}4)$$

$p = \text{Prob}\,[m(t) = 1]$.

For dc offsets less than the signal amplitude (the only case of any practical interest) the error signal is not changed markedly from (3-1). For negligible offsets, quadrature errors and amplitude imbalance, the characteristic is seen to be proportional to the signal amplitude for the limited case and proportional to the signal power for the true multiplier case. Similar problems exist with D_i and D_q and quadrature imbalance. However, as with the previous case, with the assumption of adequate transition density or with reference phase inversions, capacitive coupling renders the offsets negligible and the quadrature error can be controlled to insignificant levels. The results for this case are completely described in Appendix C with the effects of degradation shown. This implementation, however, does not suffer from any significant hardware degradations since the circuits can be designed with insignificant dc offsets, quadrature errors, and amplitude imbalances.

3.2 HAA Performance

The test results of the HAA with the configuration shown in Figure 3-1 (except analog multiplier is used in place of the limiters) are presented in the following paragraphs. For these tests the noise bandwidth of the I and Q acquisition channels was established at 50 kHz and the HAA phase characteristic was chosen with $F_1 = 100$ Hz and $F_2 = 50$ kHz as shown in Figures C-7 and C-8 of Appendix C. The ST/N_0 calibration was performed as discussed in Appendix C. Each set of data is presented for two data rates, R = 100 bps and R = 200 bps. As shown by these results, the data rate has no effect on the acquisition performance, as was expected [(3-1) through (3-3)].

Typical acquisition trajectories for the three S/N_0 test values and for initial frequency offsets [$\Delta F(o)$] of 44.8 kHz and 11.2 kHz for R = 100 bps are given in Figures 3-2 and 3-3. As seen in these figures, the trajectories exhibit a damped oscillation about the center frequency, with the damping coefficient varying with S/N_0. An AGC algorithm could be added to the algorithm if constant damping is desired. Typical trajectories are not given for R = 200 bps because of their similiarity to those for R = 100 bps.

3.2.1 PERFORMANCE TEST RESULTS

The test data for data rates, R = 100 and 200 bps, respectively, are shown in Tables 3-1 and 3-2. These tables include statistical data from 64 separate acquisition trials at each of three frequency offsets, the signal-to-noise densities (S/N_0), and two data rates. The test statistics were maximum absolute offset frequency, MAX $|\Delta F(t)|$; average absolute frequency offset $\overline{|\Delta F(t)|}$, and average frequency offset $\overline{\Delta F(t)}$. The data from these tables is also plotted in Figures 3-4 through 3-6. The statistical significance of the maxima shown in Figure 3-4 is questionable. However, since true variances cannot be calculated with the present test arrangement this parameter is presented. The average absolute frequency offset of Figure 3-5 is an estimate of the variance. This curve in conjunction with the average offset in Figure 3-6 can be used to approximate the frequency jitter at any given time. However, it must be realized that jitter is not of primary importance since this is an acquisition technique, not a tracking scheme. Of primary importance would be the mean time of the first zero crossing and its derivative at the crossing. This is the

case because the lock detector detects the signal quickly and with high probability as it passes into the narrow filters of the tracking channels. When detected the HAA either is disabled or much reduced in gain to provide adequate frequency tracking performance.

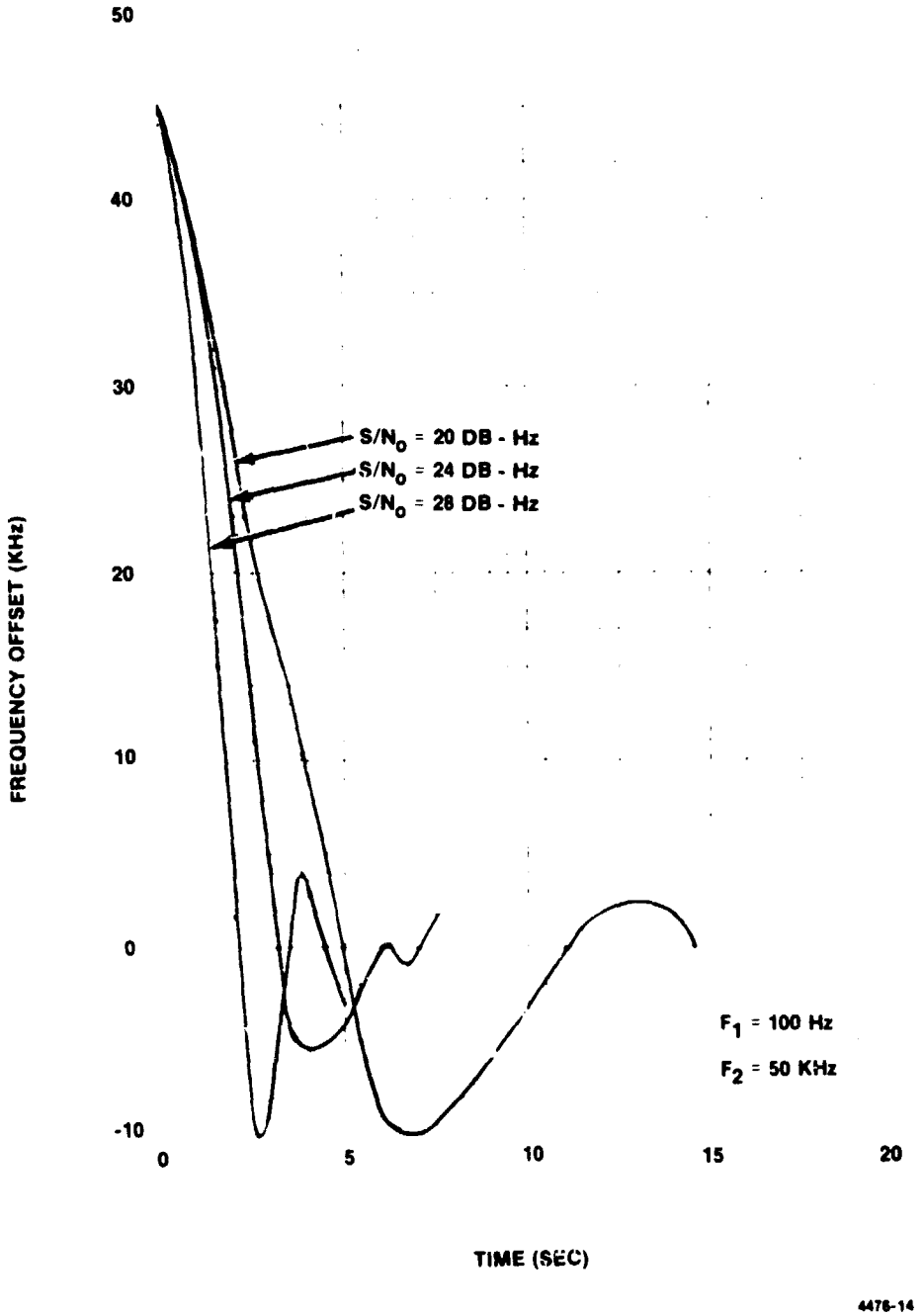


Figure 3-2. Typical Acquisition Trajectories, $\Delta f(0) = 44.8$ kHz

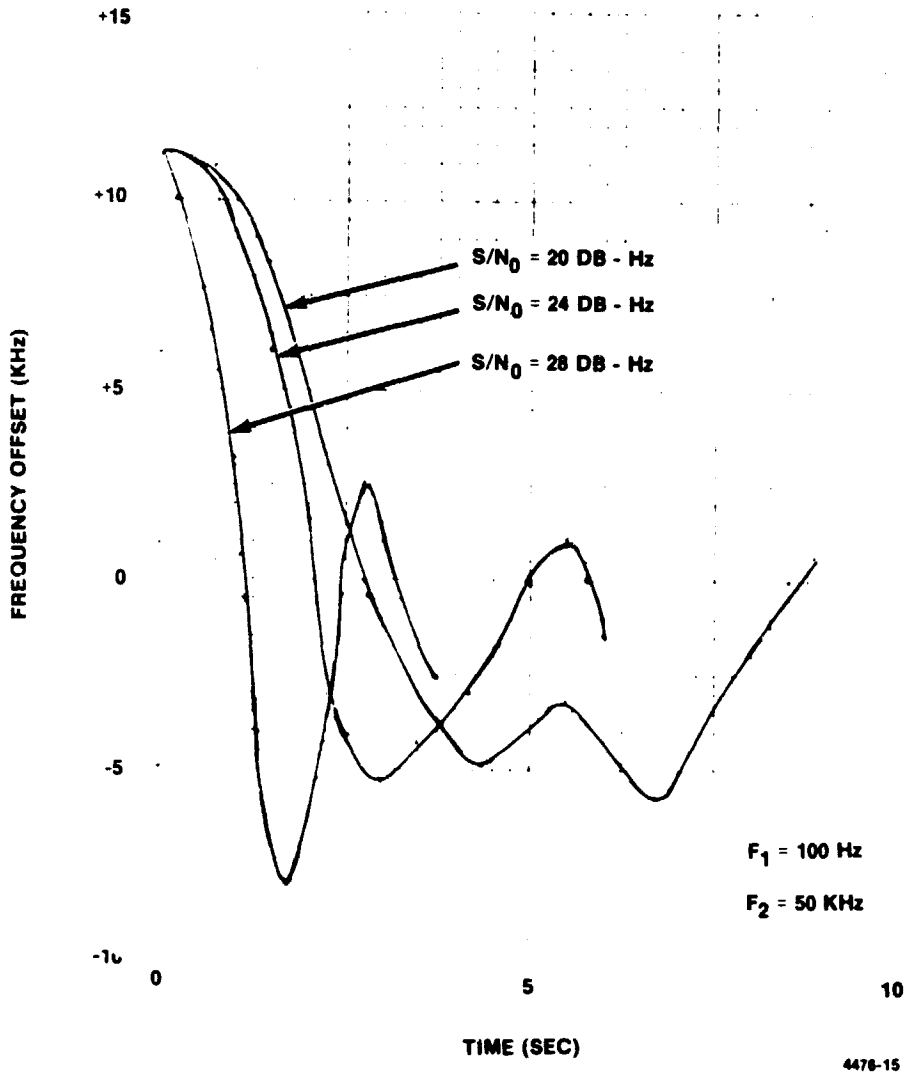


Figure 3-3. Typical Acquisition Trajectories, $\Delta f(0) = 11.2$ kHz

ORIGINAL PAGE IS
OF POOR QUALITY

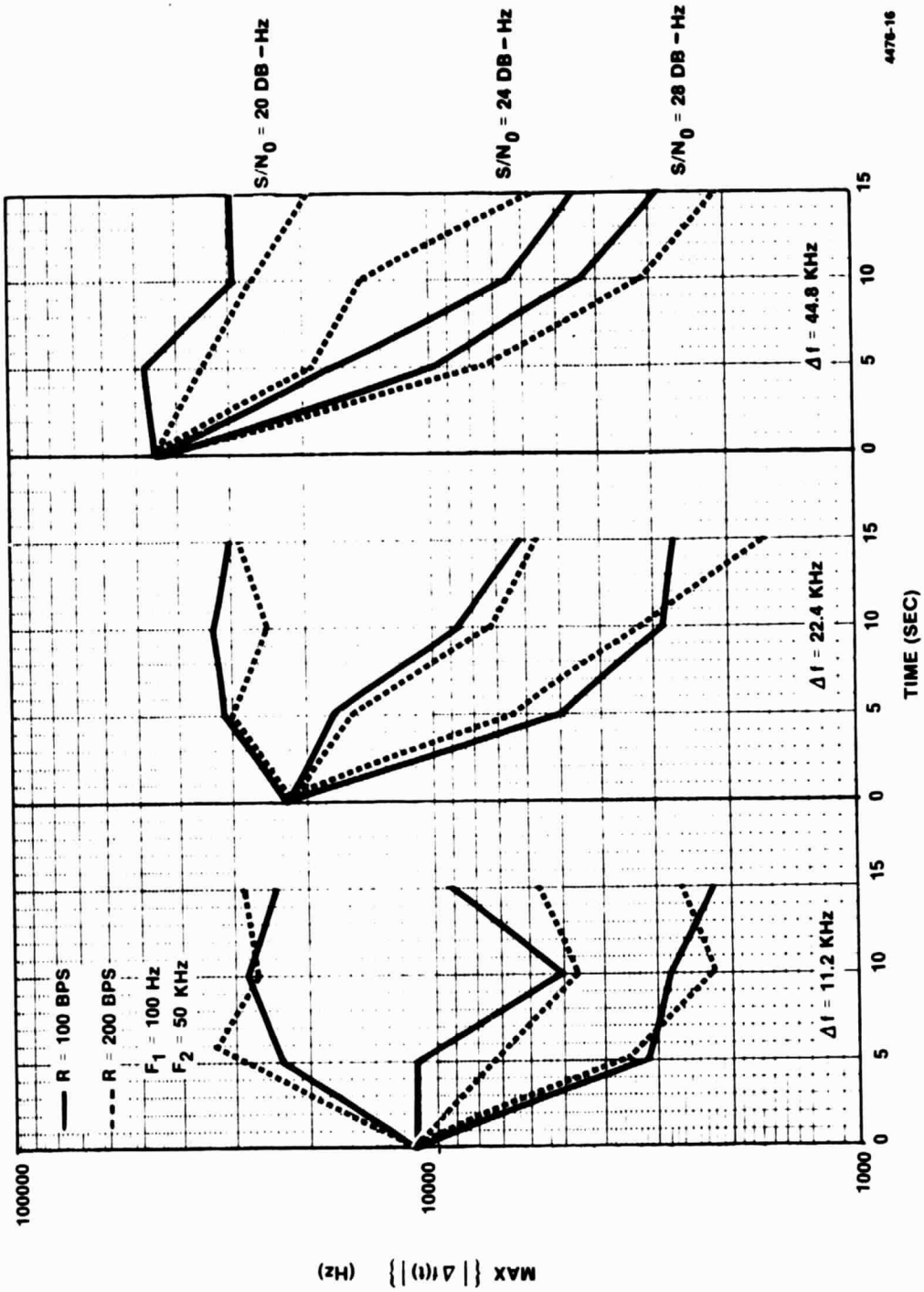


Figure 3-4. Maximum Frequency Offset

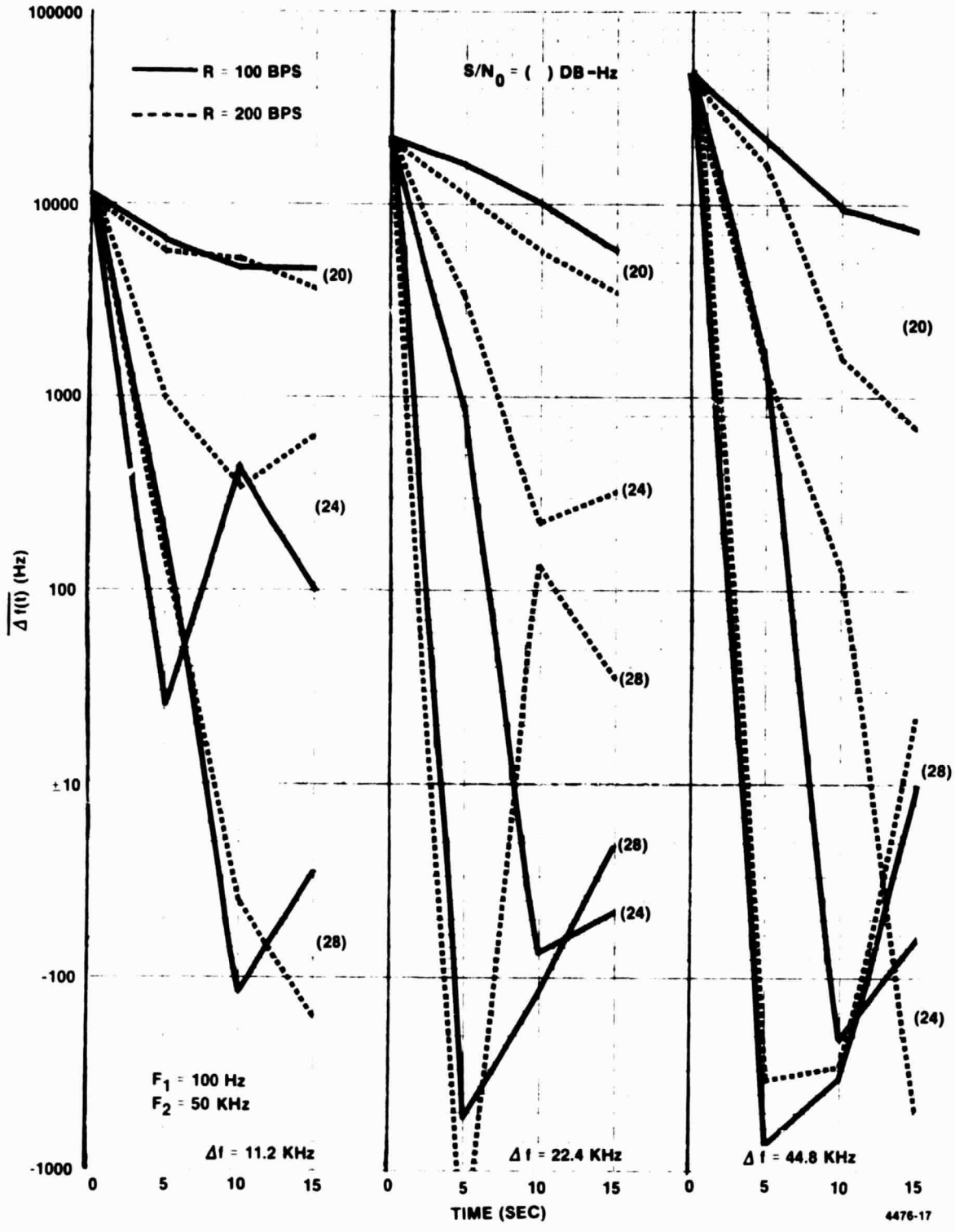


Figure 3-5. Average Frequency Offset

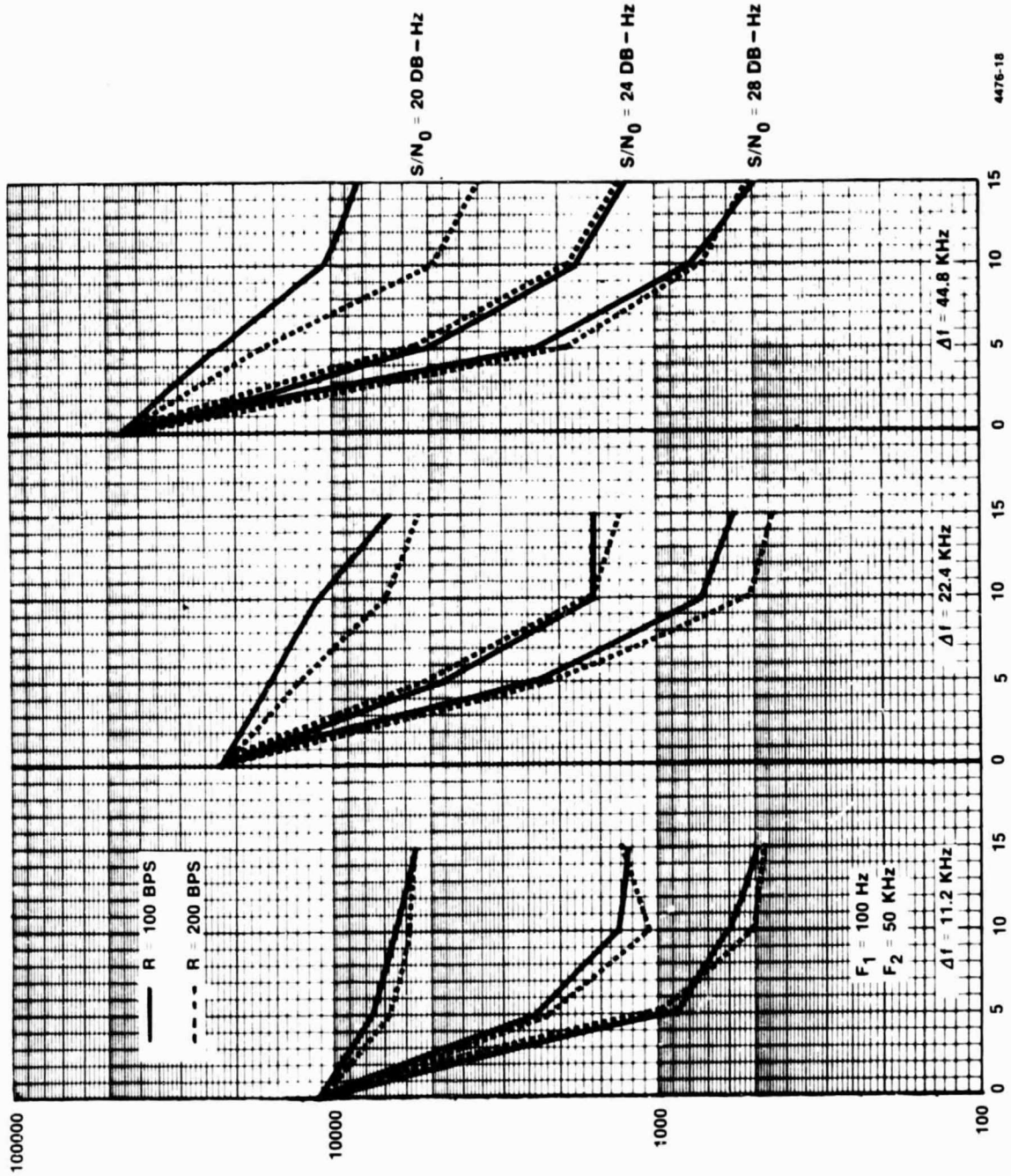


Figure 3-6. Average Absolute Frequency Offset

For the data rate R = 200 bps, the data is presented on the same plots for easy comparisons. Typical trajectories were not shown for R = 200 bps because of the similarity to R = 100 bps. Demonstrated by these results is the lack of effect of the modulation as expected from the results of Appendix C. Only S/N_0 affects the performance.

Table 3-1. HAA Test Results (R = 100 bps, F_1 = 100 Hz, F_2 = 50 KHz) for 64 Acquisition Trials

		S/N_0 = 20 dB-Hz			S/N_0 = 24 dB-Hz			S/N_0 = 28 dB-Hz		
$\Delta f(0)$	t(s)	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$
4.48 kHz	5	47828	22384	22213	17129	5193	1590	9668	2330	-718
	10	28751	10751	9686	6484	1765	-208	4300	784	-336
	15	29339	8410	7272	4489	1229	-66	2874	503	7
22.4 kHz	5	31345	15776	15335	17516	4584	846	4974	2352	-532
	10	33212	11075	10357	8636	1546	-73	2848	718	-120
	15	29995	6659	5707	6160	1535	-47	2684	576	-21
11.2 kHz	5	23307	7739	6506	11075	2370	26	3187	875	182
	10	27985	6510	4690	5040	1317	452	2797	598	-120
	15	24131	5638	4672	9154	1243	98	2221	507	-29

Table 3-2. HAA Test Results (R = 200 bps, F_1 = 100 Hz, F_2 = 50 kHz)

		S/N_0 = 20 dB-Hz			S/N_0 = 24 dB-Hz			S/N_0 = 28 dB-Hz		
$\Delta f(0)$	t(s)	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$	Max$\lvert\Delta f(t)\rvert$	$\overline{\lvert\Delta f(t)\rvert}$	$\overline{\Delta f(t)}$
44.8 kHz	5	34908	16414	16378	19124	5558	1349	7370	1882	-350
	10	26699	4916	1645	14445	1838	146	3093	715	-281
	15	19303	3512	686	5682	1280	-503	2112	514	22
22.4 kHz	5	30331	12822	11783	15517	5153	3366	6521	2294	-2133
	10	24576	6863	5733	7713	1586	219	3362	514	142
	15	28471	5423	3534	[illegible]	1306	332	1634	427	36
11.2 kHz	5	28672	6794	5660	15386	2181	992	3413	908	171
	10	26487	5879	5109	4635	1058	357	2232	514	-40
	15	28442	5656	3643	5715	1294	631	2604	485	-164

3.2.2 CONCLUSION

This test data is considered verification of the HAA concept. Because of the fledgling nature of this technique and the shortness of development time, it is not believed to be performing to its optimum at this stage. Furthermore, it must be realized that since this is an acquisition technique, a lock detector and more extensive testing is required in order to characterize the full receiver acquisition performance. Because it is likely that the tracking algorithms would be operating concurrently with the HAA algorithm for the actual receiver, the acquisition is expected to be essentially determined by the first zero crossing of the frequency acquisition trajectory.

Continuing development is required to test the preferred limiting version of the HAA and to adequately characterize the acquisition probability. However, the results to date indicate that acquisition times of approximately 5 seconds at the design S/N_0 of 24 dB-Hz are achievable. As the data rate is increased, acquisition improves for the same ST/N_0, since the data rate plays no part in the acquisition process and only S/N_0 affects performance.

SECTION 4

4. DEMODULATION

The Motorola design for the demodulation of the input signal is shown in Figure 4-1. This scheme was selected to maximize the bit error probability performance of the Hybrid Receiver at the expense of processing complexity. Because the communication environment for the JOP mission is difficult and the transmitter power is constrained, the margin from theoretical performance allocated to the receiver must be kept at a minimum. The processing power of the circuits developed for the NASA Standard CDU is more than adequate for this task. Coupled with an ADC, an NCO, and pre-accumulators, the Motorola-designed processor can perform the demodulation of signals with a minimum of degradation from theoretical performance.

In the paragraphs that follow, the carrier tracking loop and bit synchronization loop are analyzed; both theoretical and experimental performance are given.

4.1 Carrier Tracking Loop

The Hybrid Receiver uses a baseband sampling scheme to implement a second-order, suppressed-carrier, data-aided loop. The many advantages of the data-aided configuration have been well documented[1,2].

The carrier tracking algorithm is shown schematically in Figure 4-1. The in-phase samples (I_i) and the quadrature samples (Q_i) are summed in integrate-and-dump accumulators. With nominally 160 samples per bit interval and 8-bit quantization in the ADC, these accumulators can be considered to approximate perfect reset integrators. The outputs of these accumulators are applied to an arc tangent mapping ROM with a resolution of $2\pi/1024$ radians. This estimate of the phase error is applied to the loop filter which drives the phase of a NCO to achieve synchronization. The ADC sample instants are determined by the Ultra-Stable Oscillator (USO) frequency and phase, while the NCO translates the input frequency and shifts the input phase to effect tracking.

This algorithm offers the advantage of coherent tracking performance when the channel phase characteristic is approximately constant over a bit interval. Where the channel phase characteristic is approximately constant over two bit intervals the differential detection scheme shown in the figure can be shown to be the optimum a posteriori receiver of differentially encoded data.[3]

The entire tracking algorithm is stored in ROM, with ROM-resident loop coefficients selected to minimize steady-state phase jitter. This section describes the detailed design, analysis, and implementation of the Hybrid Receiver carrier tracking loop.

Simon, M.K., and Springett, J.C., "The Theory, Design, and Operation of the Suppressed Carrier Data-Aided Tracking Receiver," Jet Propulsion Laboratory, TR 32-1583, June 15, 1973.

Lindsey, W.C., and Simon, M.K., "Telecommunication Systems Engineering," Prentice-Hall Inc., 1973, pp. 530-546.

Lindsey, W.C., and Simon, M.K., op. cit., pp. 246-253.

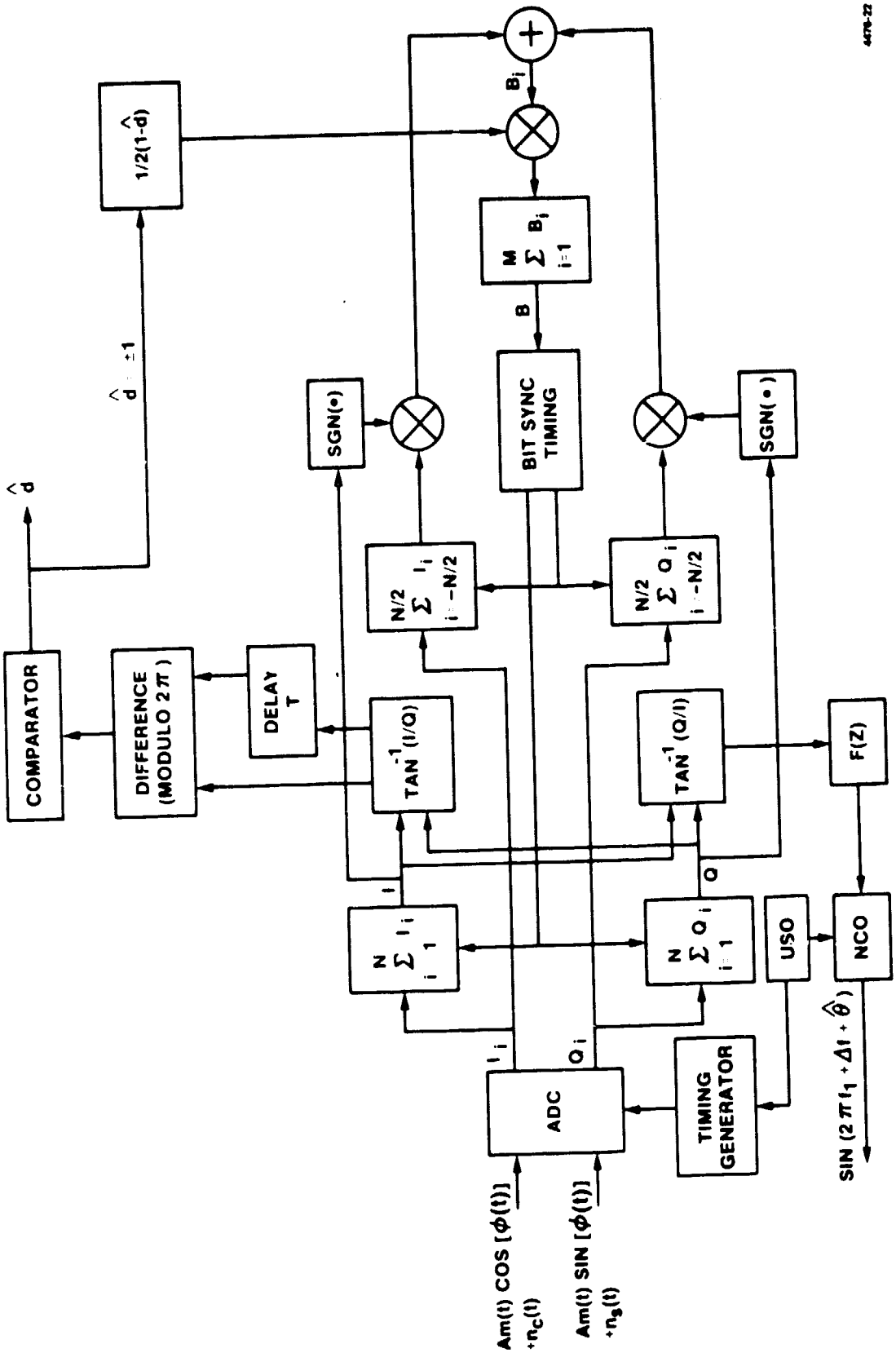


Figure 4-1. Differentially Coherent Map Detector For Differentially Encoded BPSK

ORIGINAL PAGE IS
OF POOR QUALITY

Current investigations are in progress to develop a coherent sampling[4] scheme to implement the HAA which would then permit IF sampling for the tracking algorithms similar to the approach used in the NASA Standard CDU.[5]

4.1.1 CARRIER LOOP MODEL

A simplified model for the carrier tracking loop is shown in Figure 4-2. The significant components are described below:

Quantizer-Sampler — This is the model of the ADC which has 8-bit quantization of the input signal over a range of $\pm V_v$ volts and holds each value until another sample is taken. The ADC digitizes the I and Q baseband channels which have a signal component

$$S_I = Am(t) \cos \theta \tag{4-1}$$

$$S_Q = Am(t) \sin \theta \tag{4-2}$$

The value of the digital number out of the sampler is

$$I = \left[\frac{(2^{n-1} - 1)}{V_v} S_I\right] \tag{4-3}$$

$$Q = \left[\frac{(2^{n-1} - 1)}{V_v} S_Q\right] \tag{4-4}$$

where the brackets indicate the integer part

N-Sample Accumulate and Dump — This operator simply accumulates the sampler output values over a data-rate-dependent number of samples, then dumps the accumulated value into the loop filter and clears itself.

Arc Tangent — This function is performed by a read only memory which accepts as its input the I and Q values, scales them appropriately and outputs a signed digital number proportional to the phase error. Because this element performs its own scaling it removes the effects of dc gains and signal levels at its input and allows the phase estimate to be independent of input levels as well as modulation polarity.

Loop Filter — The carrier loop filter is a first order digital filter consisting of two parallel branches (Appendix B). One branch simply scales the input; the second branch is a scaled running accumulation of the input. The filter output is the sum of both branch outputs.

NCO — This oscillator accepts a digital control word from the output of the loop filter which either advances or retards the phase of its output frequency.

Although the model of the carrier tracking loop is conceptually simple, there is an area where a simplifying assumption appears to be necessary to render the loop equation analytically tractable. This is the clocking of the output of the accumulate-and-dump operator at a sub-multiple of the sample clock.

---

[4] Springett, J.C., "Coherent Demodulation by Sampling," Jet Propulsion Laboratory, IOM 3300-75-025, February 13, 1975.

[5] Klare, S.W., and McCallister, R.D., "NASA Standard Command Detector Unit Engineering Report," February 28, 1977, Motorola, Government Electronics Division, Scottsdale, Arizona.

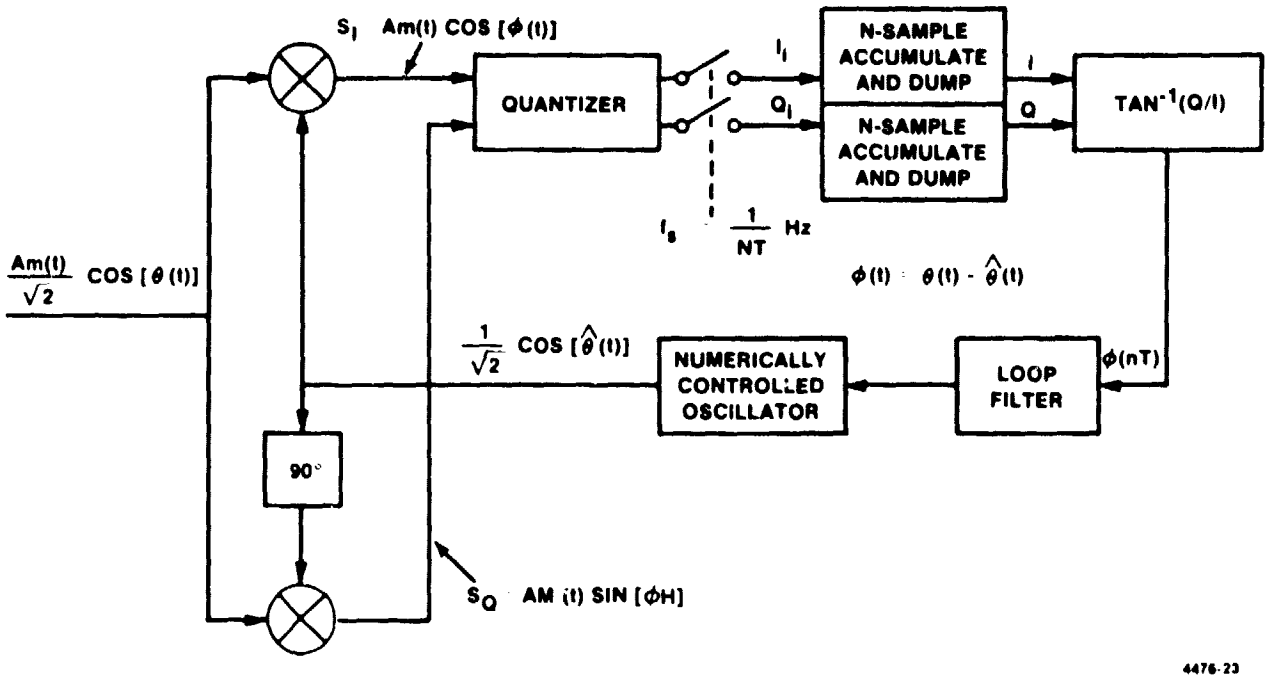


Figure 4-2. Carrier Loop Model

This difficulty is not easily accommodated. It is relatively easily shown (Appendix A) that the formal transform of the accumulate-and dump system component is not defined. This is, of course, a serious setback to the intended z-transform approach to the loop performance analysis.

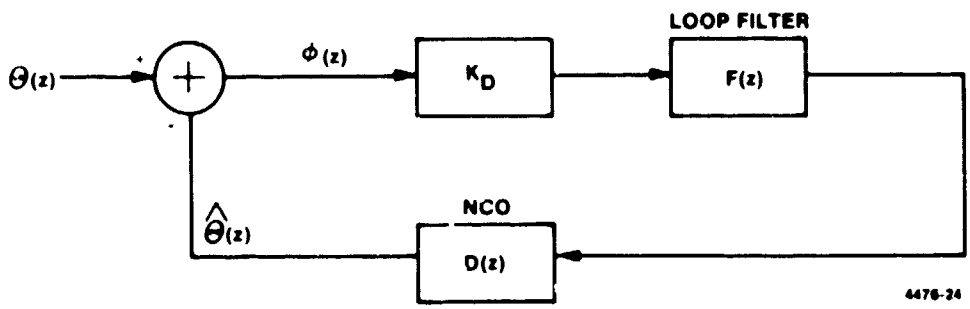
In general, there is no satisfactory simplifying assumption that is applicable to the N-sample accumulate-and-dump. However, in the special case in which all other significant system time-constants are much smaller than that associated with the accumulate-and-dump, we may invoke a discrete system version of the "delta-correlated" argument. That is, since the accumulate-and-dump corresponds roughly to a low-pass filter with the corner frequency at the data frequency, its effect on all spectral components at the much lower frequencies important to loop dynamics can be approximately modeled by multiplication by the dc-gain (N).

Assumption: The effect of the accumulate-and-dump elements may be modeled by preceding the carrier tracking loop input with the unit-dc-gain associated attenuation characteristic, leaving the dc-gain (N) to contribute to the open-loop gain of the carrier tracking loop. However, because the accumulate and dump filters are followed by the arc tangent function which is insensitive to input levels their effect will be ignored completely.

4.1.2 CARRIER LOOP ANALYSIS

4.1.2.1 Z-Transform of Closed-Loop System

Neglecting for the moment the impact of the matched filter (accumulate-and-dump) the carrier tracking loop model becomes as shown in Figure 4-3 below.



$\Theta(z)$	INPUT PHASE
$\hat{\Theta}(z)$	AVERAGE OF PHASE SAMPLES OVER ACCUMULATION PERIOD
$\phi(z)$	CARRIER TRACKING LOOP PHASE ERROR
K_D	DETECTOR GAIN (DIGITAL UNITS/RADIAN)
	512/π DIGITAL UNITS/RADIAN
T	BIT PERIOD
F(z)	LOOP FILTER TRANSFER FUNCTION
D(z)	NUMERICALLY CONTROLLED OSCILLATOR TRANSFER FUNCTION

Figure 4-3 Simplified Carrier Loop Model

The z-transforms of the loop components are easily evaluated in Appendix B yielding

$$F(z) = E_0 + E_1 \frac{z}{z - 1} \tag{4-5}$$

$$D(z) = \frac{K\phi}{z - 1} \tag{4-6}$$

where

E_0 = First-order loop coefficient

E_1 = Second-order loop coefficient

K_ϕ = NCO gain (radians/digital count)

= $\pi/1024$ radians/digital count

Then the open-loop transfer function is simply

$$G(z) = K \frac{F(z)}{z - 1} \tag{4-7}$$

where

$$K = K_0 K_\phi \tag{4-8}$$

The desired loop transfer function is given by

$$\frac{\hat{\theta}(z)}{\theta(z)} = H(z) = \frac{G(z)}{1 + G(z)} \tag{4-9}$$

or

$$H(z) = \frac{KF(z)}{(z - 1) + KF(z)} \tag{4-10}$$

Upon substitution of (4-5) into (4-10), the closed loop transfer function simplifies to

$$H(z) = \frac{K(E_0 + E_1) z - (K - E_0)}{(z - 1) + K(E_0 + E_1) z - (K - E_0)} \tag{4-11}$$

4.1.2.2 Analog Equivalent Loop Transfer Function

It would facilitate the procedure for obtaining optimal receiver performance if we could draw upon the wealth of analysis and experience with analog tracking loops by deriving the *equivalent* analog loop equations. Assume that the loop dynamics are primarily dictated by input spectral components much lower than either the sampling frequency or data frequency (i.e., $f << 1/T$).

Thus

$$\omega T \ll 1 \tag{4-12}$$

But

$$z = \text{EXP}(sT) = 1 + sT + \frac{1}{2}(sT)^2 + \ldots \tag{4-13}$$

Which implies that the loop equation may be approximated by substituting for the variable z in terms of s.

$$z \simeq 1 + sT \tag{4-14}$$

To this end, define the *equivalent analog* carrier tracking loop transfer function

$$\frac{\hat{\Theta}(s)}{\Theta(s)} \triangleq H(s) \triangleq H(z) \Big|_{z = 1 + sT} \tag{4-15}$$

Then

$$H(s) = \frac{(2\zeta\omega_n)\, s + \omega_n^2}{s^2 + (2\zeta\omega_n)\, s + \omega_n^2} \tag{4-16}$$

where

$$\omega_n = \sqrt{\frac{1}{T} K \cdot E_1} \tag{4-17}$$

$$\zeta = K\left(\frac{E_0 + E_1}{T}\right) \Big/ (2\omega_n) = \frac{K(E_0 + E_1)}{2\sqrt{K \cdot E_1}} \tag{4-18}$$

4.1.3 CARRIER TRACKING LOOP DESIGN POINT

Having (4-16) the loop operating point may be set in the following systematic fashion:

- Determine the manner in which all data-rate dependent coefficients must vary in order to maintain constant carrier tracking loop jitter at threshold over all data rates.
- Determine an appropriate carrier loop operating point and the corresponding loop dynamics.

4.1.3.1 Data-Rate Dependent Coefficients

The carrier tracking loop noise bandwidth may be approximately determined by using the analog equivalent loop transfer function.[1]

$$B_L = \left(\frac{\omega_n}{2}\right)\left(\zeta + \frac{1}{4\zeta}\right) = \left(\frac{4}{T}\right)\left(\frac{1}{4}\right)\left[K(E_0 + E_1) + \left(\frac{E_1}{E_0 + E_1}\right)\right] \tag{4-19}$$

---

[1] Lindsey, W.C., "Synchronization Systems in Communication And Control," Prentice-Hall, Englewood Cliffs, New Jersey, 1972

Then, since the carrier loop tracking coefficients are constant over all data rates, simply assuring that K, the tracking loop dc gain, remains constant over all data rates guarantees that the primary design criterion will be met. The dc gain K for this design is

$$K = K_D K_\phi = \frac{512}{\pi} \cdot \frac{\pi}{1024} = \frac{1}{2} \tag{4-20}$$

4.1.3.2 Determination of Carrier Tracking Loop Operating Point

The second phase of the carrier tracking loop design involves two operations:

- Derivation of coefficient values which result in tracking performance consistent with the communications environment. This will generate a set of tentative values, whose accuracy is contingent upon the adequacy of the analog equivalent loop as a model for the discrete tracking loop behavior.
- Rigorous calculation of the carrier tracking loop noise bandwidth corresponding to the coefficient values chosen in paragraph 4.1.3.1.

Because the minimum S/N_0 is specified as 24.5 dB Hz with a minimum data rate of 100 bps, the carrier tracking loop bandwidth (B_L) must be on the order of 5 Hz to ensure linear operation of the loop and to minimize loop tracking error. Choose a target loop damping factor of

$$\zeta \approx \frac{1}{\sqrt{2}} \tag{4-21}$$

Substituting this value in (4-18) and solving for E_0 yields,

$$E_0 = E_1 \pm 2\sqrt{E_1}$$

where, since E_1 is less than unity and E_0 must be positive, only one solution is possible,

$$E_0 = E_1 + 2\sqrt{E_1} \tag{4-22}$$

Solving (4-19) for E_0 with $B_L = 5$ Hz yields,

$$E_0 = 0.2 - E_1 \pm \sqrt{0.04 - 2E_1} \tag{4-23}$$

Equating these two solutions and solving for E_1 results in the polynominal

$$X^4 + 2X^3 + 1.3X^2 - 0.2X = 0 \tag{4-24}$$

where $X = \sqrt{E_1}$

The only relevant root of this polynominal is

$$X = 0.2732, \text{ or } E_1 \approx 0.0162 \tag{4-25}$$

Since E_1 must be a binary fraction ($1/2^j$), choose

$$E_1 = 0.015625 = 2^{-6} \tag{4-26}$$

Then, substituting this value into (4-22) and solving for E_0 yields

$$E_0 = 0.265625 \approx 0.25 = 2^{-2} \quad (4\text{-}27)$$

Using these values in (4-18) and (4-19) results in a loop bandwidth and damping factor of

$$B_L = 4.79 \text{ Hz} \quad (4\text{-}28)$$

$$\zeta = 0.7513 \quad (4\text{-}29)$$

These values have been used in the Hybrid Receiver breadboard with acceptable performance as discussed in paragraph 4.3. It should be noted that these coefficients have been selected on the basis of S/N_0 and damping factor. The dynamics of 50 Hz/s will require a third-order loop for acceptable performance, or if the 50 Hz/s Doppler rate is only transitory, more information on its characteristics is required to select an appropriate loop filter configuration.

4.1.3.3 Bit Synchronization Loop Components

Sampler/Quantizer — This element is identical to that described in paragraph 4.1.

In-Phase Accumulator — Data samples are accumulated over the estimated bit duration to implement a digital matched filter for both I and Q channels.

Mid-Phase Accumulator — Data samples are accumulated over one bit duration starting from each estimated mid-bit epoch for both I and Q channels.

SGN(•) — Following the end-of-bit, the received data bit is estimated to be equal to the sign bit of the in-phase digital accumulator.

Transition Detector — Clearly the mid-phase integration contains no estimate of bit synchronization timing error unless the data changed polarity. The transition detector establishes the proper weighting for the mid-phase accumulation value. For this case, with differential encoding, the detection of a -1 indicated a phase inversion of the input.

Bit-Synchronization Accumulator — The properly weighted mid-phase accumulator values are accumulated over M data transitions to provide a narrow bit synchronization tracking bandwidth.

Scaling — Data-rate dependent scaling provides identical bit synchronization loop tracking dynamics scaled to the input data rate.

BS Clock — The bit-synchronization clock is advanced/retarded as a function of the input correction word to move the estimated end-of-bit to within the nearest subcarrier cycle. This limited resolution is then augmented by the subcarrier tracking loop performance to provide adequate overall bit-synchronization resolution.

4.2 Bit Timing Synchronization Loop

Proper detection of the demodulated data stream requires the use of some form of matched filter. Implementation of this matched filter, in turn, requires the generation of accurate estimates of the end-of-bit epoch. It is the function of the bit synchronization loop to generate these end-of-bit estimates.

The bit synchronization configuration, a digital data-transition tracking loop, consists of two parallel branches which are strobed by a timing generator driven by an error signal formed from the product of the branch outputs. The in-phase branch monitors the polarity of the actual transitions of the input data and the mid-phase branch obtains a measure of the lack of synchronization.

The performance of this loop has been the subject of extensive analysis,[7,8,9] at NASA's Jet Propulsion Laboratory. These results will be drawn on throughout this section, which describes the detailed design and implementation of the Hybrid Receiver bit synchronization loop.

4.2.1 BIT SYNCHRONIZATION LOOP MODEL

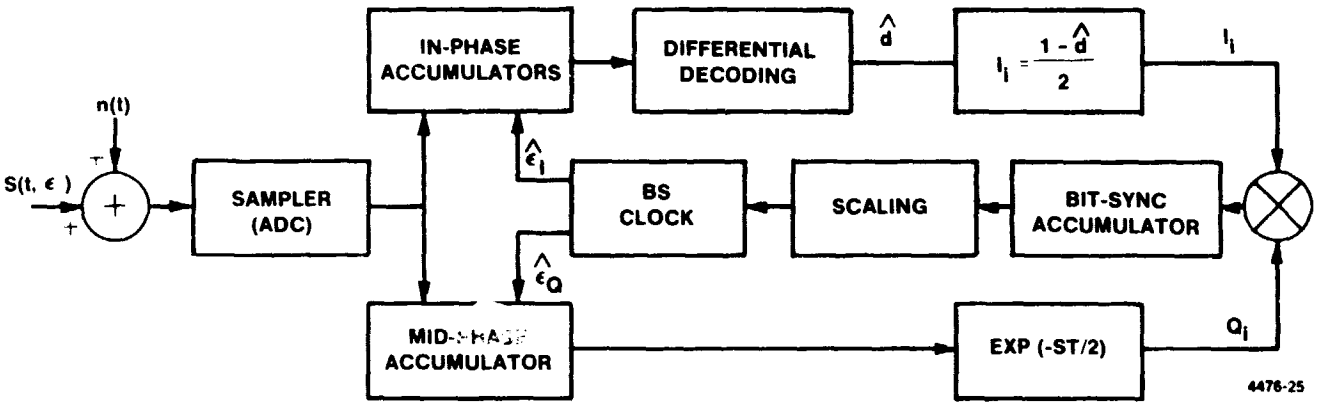


Figure 4-4. Bit Synchronization Loop Model

4.2.2 ASSUMPTIONS

There are two fundamental assumptions required to facilitate the analysis of the bit synchronization tracking loop:

- Bit synchronization tracking error is slowly varying relative to the data rate.
- The approach employed in analyzing the carrier tracking loop may be used to resolve difficulties due to clocking the accumulate-and-dump operators at sub-multiples of the input clock rate.

---

[7] Lindsey, W.C., and Tausworth, R.C., "Digital Data-Transition Tracking Loops," JPL Space Programs Summary 37-50, Volume III, April 30, 1968, pp. 272-276.

[8] Simon, M.K., "An Analysis of the Steady-State Phase Noise Performance of a Digital Data-Transition Tracking Loop," JPL Space Programs Summary 37-55, Volume III, February 28, 1969, pp. 54-62.

[9] Simon, M.K., "Optimization of the Performance of a Digital-Data-Transition Tracking Loop," IEEE Transactions on Communication Technology, October 1970, pp. 686-689.

ORIGINAL PAGE IS
OF POOR QUALITY

The significance of the first assumption is that it permits us to model the bit synchronizer as a continuous phase-lock loop.[10,11] With this step, we may take advantage of the thorough analysis of this type of bit synchronizer which has been previously performed.[12]

The significance of the second assumption lies in its effect on the bit loop integrator. As with the subcarrier loop, we replace the accumulator by its dc gain (M), and consider the system clock time to be reduced by a factor of M. Note that this assumption further constrains the previous one. The bit synchronization tracking error must be slowly varying relative to the data rate divided by a factor of M.

4.2.3 ANALYSIS[10]

The digital data-transition tracking loop may be modeled in terms of the generalized tracking loop shown in Figure 4-5.

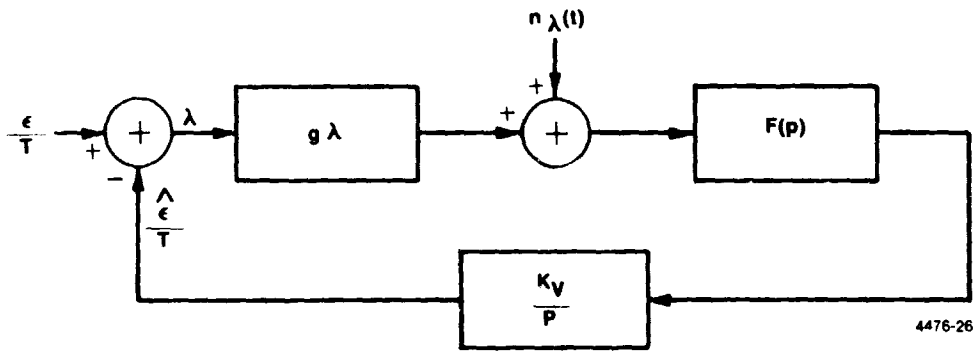


Figure 4-5. Simplified Bit Synchronization Loop Model

[10] Simon, M.K., "An Analysis of the Steady-State Phase Noise Performance of a Digital Data-Transition Tracking Loop," JPL Space Programs Summary, 37-55, Vol. III, pp. 55-62.

[11] Simon, M.K., "Optimization of the Performance of a Digital-Data-Transition Tracking Loop," IEEE Trans. on Comm. Technology, October 1970, pp. 686-689.

[12] Simon, M.K., and Lindsey, W.C., "Telecommunication Systems Engineering, Prentice-Hall Inc., 1973, pp. 442-457.

where

ϵ = Input epoch

$\hat{\epsilon}$ = Local estimate of input epoch

T = Bit period

λ = Normalized bit synchronization timing error

$g(\lambda)$ = Equivalent loop nonlinearity

$n\lambda(t)$ = Equivalent timing noise process

F(p) = Loop filter (differential operator form)

K_v = Gain of digital VCO

p = Heaviside operator

The stochastic differential equation which describes the loop operation is then

$$\dot{\lambda} = -K_v F(p) [g(\lambda) + \eta_\lambda(t)] \tag{4-30}$$

Evaluation of $g(\lambda)$ and the expression for the two-sided noise spectral density of the equivalent noise process are required for solution of this equation. The derivation is somewhat involved, so only salient results are shown.

The equivalent loop nonlinearity is given by

$$g(\lambda) = K_2 AT \bullet \left[\lambda \operatorname{erf} \sqrt{R}\,(1-2\lambda) - \frac{1}{8}(1-2\lambda)\right]\left\{\operatorname{erf}(\sqrt{R}) - \operatorname{erf}\left[\sqrt{R}\,(1-2\lambda)\right]\right\} \tag{4-31}$$

where

A = signal amplitude

K_2 = midphase branch gain

and the two-sided noise spectral density (evaluated at zero)

$$S(0,\lambda) = K_v^2 \left(\frac{N_0 T}{4}\right) h(\lambda) \tag{4-32}$$

where

$$H(\lambda) = 1 + \frac{R}{2}\left(1 + 12\lambda^2\right) \tag{4-33}$$

$$- \frac{3}{8\pi}\left\{\exp^2\left[-R\,(1-2\lambda)^2\right] + \exp^2\left[-R\right]\right\} \tag{4-34}$$

$$- \frac{1}{4\pi}\exp\left(-R\right)\exp\left[-R\,(1-2\lambda)^2\right]$$

$$+ \frac{\mathrm{erf}^2\left[\sqrt{R}\,(1-2\lambda)\right]}{16}\left[4 - R\,(12\lambda^2 + 20\lambda + 3)\right]$$

$$+ \frac{\mathrm{erf}^2\left(\sqrt{R}\right)}{16}\left\{-4 - 12\,R\left[\left(\lambda - \frac{1}{2}\right)^2\right]\right\}$$

$$- \frac{\mathrm{erf}\left(\sqrt{R}\right)\mathrm{erf}\left[\sqrt{R}\,(1-2\lambda)\right]}{2} R\left(\lambda - \frac{1}{2}\right)^2$$

$$+ \frac{(1-2\lambda)}{8}\sqrt{\frac{R}{\pi}}\quad \exp\left(-R\right)\mathrm{erf}\left(\sqrt{R}\right)$$

$$- \frac{3(1-2\lambda)}{8}\sqrt{\frac{R}{\pi}}\quad \exp\left(-R\right)\mathrm{erf}\left[\sqrt{R}\,(1-2\lambda)\right]$$

$$- \frac{(3-6\lambda)}{8}\sqrt{\frac{R}{\pi}}\quad \exp\left[-R\,(1-2\lambda)^2\right]\mathrm{erf}\left[\sqrt{R}\,(1-2\lambda)\right]$$

$$- \frac{3+10\lambda}{8}\sqrt{\frac{R}{\pi}}\quad \exp\left[-R\,(1-2\lambda)^2\right]\mathrm{erf}\left(\sqrt{R}\right) \text{ for } \lambda \leqslant \frac{1}{2}$$

and

$$\mathrm{erf}(X) = \frac{2}{\sqrt{\pi}}\int_0^X e^{-t^2}\,dt \tag{4-35}$$

Then the probability density function may be found by solving the stationary Fokker-Planck differential equation

$$\frac{1}{2}\frac{3}{d\lambda^2}\left[(K_v\,K_p)^2\,S(0,\lambda)\,P_s(\lambda)\right] + \frac{d}{d\lambda}\left[K_v\,K_p\,g(\lambda)\,P_s(\lambda)\right] = 0 \tag{4-36}$$

where

$K_p = F(p)$ for 1st order loop filter

Solving (4-36) yields the following steady-state probability density function (assuming strong signal conditions):

$$P_s(\lambda) = C_1 \bullet \mathrm{EXP}\left[-\int_0^{\lambda} \frac{K_2 h(y) + YR\delta_0 g(y)/(K_2 AT)}{K_2 h(y)}\, dy\right] \tag{4-37}$$

where C_1 is chosen such that

$$2\int_0^{1/2} P_s(\lambda)d\lambda = 1 \tag{4-38}$$

and

$$R = \text{Signal energy to noise-density ratio} \tag{4-39}$$

$$\delta_0 = \frac{2}{W_{L_0}T}$$

$$W_{L_0} = \frac{AK_2K_1K_p}{2} \tag{4-40}$$

The normalized mean-squared synchronization error (σ_λ^2) is given by

$$\sigma_\lambda^2 = 2\int_0^{1/2} \lambda^2 P_s(\lambda)d\lambda \tag{4-41}$$

But for our purposes, the variance may be well approximated by

$$\sigma_\lambda^2 \simeq \left(\frac{1}{2R\delta_0}\right)\left[\frac{1}{1 - 2/\delta_0}\right] \tag{4-42}$$

and the linear loop bandwidth is

$$W_L = K_F W_{L_0} \tag{4-43}$$

where

$$K_F = \mathrm{erf}\,(\sqrt{R}) - \frac{1}{2}\sqrt{\frac{R}{\pi}}\ \mathrm{EXP}\,(-R) \tag{4-44}$$

4.2.4 HYBRID RECEIVER BIT SYNCHRONIZATION LOOP PERFORMANCE

The assumptions of paragraph 4.2.2 allow us to apply directly the results of paragraph 4.2.3 to the actual bit synchronization timing loop implemented. The bit synchronization loop gain is given by

$$K = K_2 K_1 K_P K_E \tag{4-45}$$

where

K_2 = 160 Samples/Transition

$K_1 = \frac{100}{160}$ Hz/digital unit

$K_P = 8 \cdot 2^{-11}$ transition scale factor

K_E = 0.952 for $R = 10^{-4}$

or

$K = 0.372$

Thus the bit synchronization loop bandwidth is simply

$$W_1 = \frac{A K}{2} = 1.488 \text{ Hz} \tag{4-46}$$

where, $A \approx 8$ digital units/sample

and the normalized bit synchronization jitter is given by

$$\sigma_\lambda^2 = \frac{1}{2R\delta_0} \tag{4-47}$$

where

$$\delta_0 = \frac{2}{W_1 T} = 134.4 \tag{4-48}$$

and the normalized bit timing jitter at $R = 10^{-4}$ is

$$\sigma_\lambda^2 = \frac{1}{2.10^{-4} \cdot 134.4} = 1.32 \times 10^{-2} \tag{4-49}$$

or, the bit synchronization rms jitter is approximately 13 degrees.

4.3 Bit Error Probability Performance

The Hybrid Receiver was tested using the tracking coefficients derived in paragraphs 4.1 and 4.2 for the algorithms illustrated in Figure 4-1. Because the carrier tracking was essentially coherent, the results approached very closely the theoretical performance for differentially encoded coherent PSK (DEPSK). The data taken is shown in Figure 4-6. The data pattern transmitted was a 15 bit maximal length PN code. The data as collected is shown in Table 4-1. The S/N_0 for each point was calibrated using the procedure outlined in Appendix C.

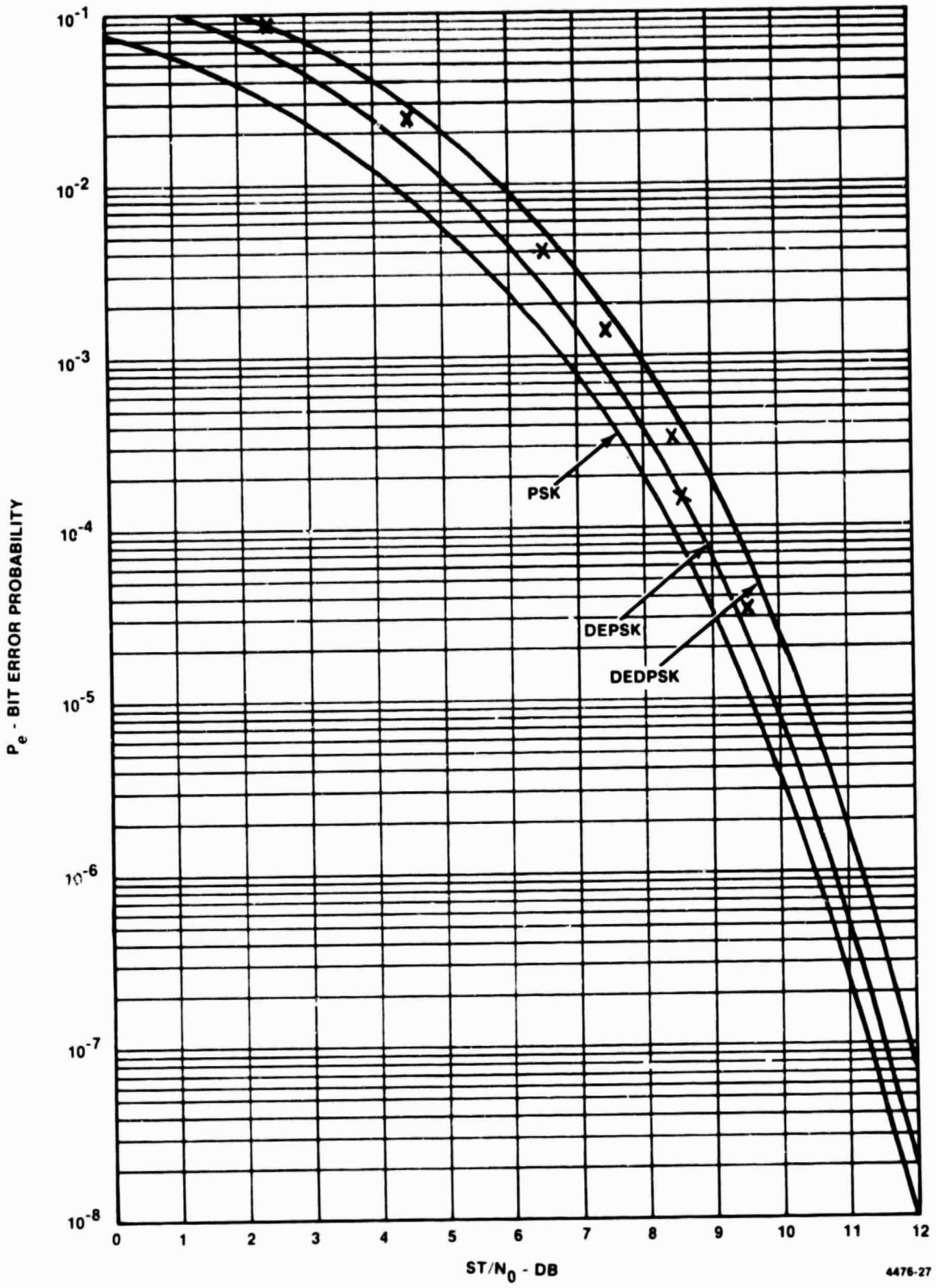


Figure 4-6. Bit Error Probability Performance

ORIGINAL PAGE IS
OF POOR QUALITY

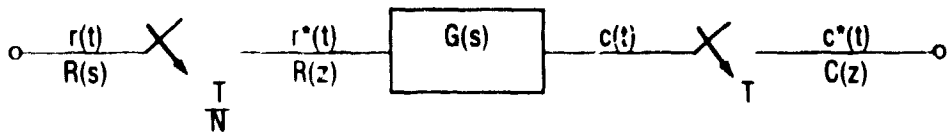
Table 4-1. Bit Error Rate Data, R = 100 bps

ST/N_0 (dB)	Total Bits Transmitted	Bit Errors	Probability of Bit Error
2.5	3.74×10^3	321	8.58×10^{-2}
4.5	1.28×10^4	323	2.52×10^{-2}
6.5	3.54×10^4	148	4.18×10^{-3}
7.5	8.80×10^4	150	1.70×10^{-3}
8.5	2.03×10^5	32	1.58×10^{-4}
9.5	7.30×10^6	234	3.21×10^{-5}

APPENDIX A
Z-TRANSFORM OF ACCUMULATE-AND-DUMP

The accumulate-and-dump component poses special difficulties for the z-transform analytic approach, since its input and output clock rates are not identical. The following analytical treatment[1] of the "multi-rate" type of discrete system will clearly indicate the nature of the difficulties and the need for a simplifying assumption.

Consider the arbitrary system G(s) whose output clock rate is a sub-multiple of its input clock rate.



Define:

$p(t)$ = Sampling Function

$$= \sum_{k=0}^{\infty} \delta\left(t - k\,\frac{T}{N}\right) \tag{A-1}$$

then

$$r^*(t) = r(t)p(t) \tag{A-2}$$

and

$$c(t) = \int_0^{\infty} r^*(\tau)g(t-\tau)d\tau \tag{A-3}$$

$$= \int_0^{\infty} \left[\sum_{k=0}^{\infty} r(\tau)\,\delta\left(\tau - k\,\frac{T}{N}\right)\right] g(t-\tau)d\tau$$

$$= \sum_{k=0}^{\infty} r\left(k\,\frac{T}{N}\right) g\left(t - k\,\frac{T}{N}\right)$$

---

[1] Kranc, George M., "Input-Output Analysis of Multirate Feedback Systems," IRE Trans. on Automatic Control, PGAC-3, November, 1957 (p. 21-28).

but

$$C(z) \triangleq \sum_{i=0}^{\infty} c(iT)z^{i} \tag{A-4}$$

$$= \sum_{i=0}^{\infty} \left[\sum_{k=0}^{\infty} r\left(k\frac{T}{N}\right) g\left(iT - k\frac{T}{N}\right) \right] z^{i}$$

and, clearly, this expression will not allow us to separate out a "transfer function," H(z), such that

$$C(z) = H(z)R(z) \tag{A-5}$$

This is not to imply that such systems will not yield to a suitable form of analysis other than that based upon a transfer function concept. To show this, we will derive an expression for the output z-transform as a reasonably simple function of the input and the system characteristic.

Decomposing the summation of a discrete function is a straightforward operation

$$\sum_{k=0}^{\infty} f(kT) = \sum_{m=0}^{N-1} \sum_{j=0}^{\infty} f\left[(jN + m)\,T\right] \qquad n = 1, 2, \ldots \tag{A-6}$$

Therefore, A-4 may be recast as

$$c(iT) = \sum_{m=0}^{N-1} \sum_{j=0}^{\infty} r\left[(jN + m)\frac{T}{N}\right] g\left[iT - \frac{T}{N}(jN + m)\right] \tag{A-7}$$

and

$$C(z) = \sum_{m=0}^{N-1} \sum_{j=0}^{\infty} r\left[(jN + m)\frac{T}{N}\right] \left[\sum_{i=0}^{\infty} g\left(iT - jT - \frac{mT}{N}\right) z^{i}\right] \tag{A-8}$$

Let

$$K = i - j \text{ in last summation} \tag{A-9}$$

Then

$$C(z) = \sum_{m=0}^{N-1} \left[\sum_{j=0}^{\infty} r\left(jT + \frac{mT}{N}\right) z^{j}\right] \left[\sum_{K=-j}^{\infty} g\left(kT - \frac{mT}{N}\right) z^{K}\right] \tag{A-10}$$

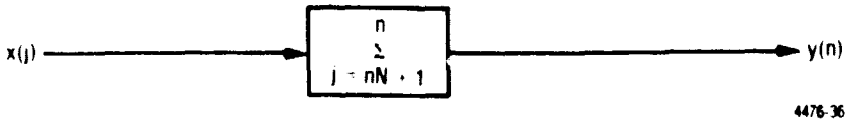
$$C(z) = \sum_{m=0}^{N-1} Z\left\{R(s)\exp\left(\frac{mT}{N}s\right)\right\} \cdot Z\left\{G(s)\exp\left(-\frac{mT}{N}s\right)\right\} \tag{A-11}$$

The analysis above corresponds to Kranc's "switch-decomposition" method, and clearly indicates the nature of the dilemma. A frontal assault on noise-bandwidth and loop dynamics calculations based upon (A-11) was considered impractical, and since attempts to construct a "pseudo-transfer function" in the sense of (A-5) met with inadequate success, the search for simplifying assumptions to facilitate the treatment of the accumulate-and-dump was initiated.

APPENDIX B
DERIVATION OF COMPONENT Z-TRANSFORMS

B.1 N-SAMPLE ACCUMULATE-AND-DUMP

This component may be implemented as an N-sample running accumulator which is simply sampled on every Nth clock. The z-transform for the N-sample running accumulator is easily determined:



Observe that

$$y(n) = \sum_{j=nN+1}^{n} x(j) \tag{B-1}$$

$$= \sum_{i=0}^{N-1} x(n-i)$$

Then

$$Y(z) = Z\left[y(n)\right] = \sum_{n=0}^{\infty}\left[\sum_{i=0}^{N-1} x(n-i)\right] z^{-n} \tag{B-2}$$

$$= \sum_{i=0}^{N-1}\sum_{n=0}^{\infty} x(n-i)z^{-n}$$

$$= \sum_{i=0}^{N-1} z^{-i} X(z)$$

$$X(z)\left(\frac{z^{N}-1}{z^{N}(z-1)}\right)$$

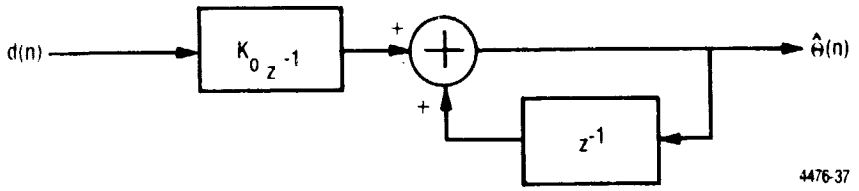
$$\therefore \quad H_N(z) = \frac{Y(z)}{X(z)} = \frac{z^{N}-1}{z^{N}(z-1)} \tag{B-3}$$

The associated dc gain can be found by taking the limit as z approaches unity

$$G_{DC} = \lim_{z \to 1} \frac{z^N - 1}{z^N - z^{N-1}} \xrightarrow{\text{L'Hospital}} \frac{N}{N - (N-1)} = N \qquad \text{(B-4)}$$

B.2 NUMERICALLY CONTROLLED OSCILLATOR

The NCO establishes the desired phase relationship with the input sinusoid by incrementing/decrementing a phase control register by the scaled output of the digital loop filter. This operation is modelled below:



The sampling phase is governed by the following equation

$$\hat{\theta}(n) = \hat{\theta}(n-1) + K\phi d(n-1) \qquad \text{(B-5)}$$

Or, in a z-transform notation

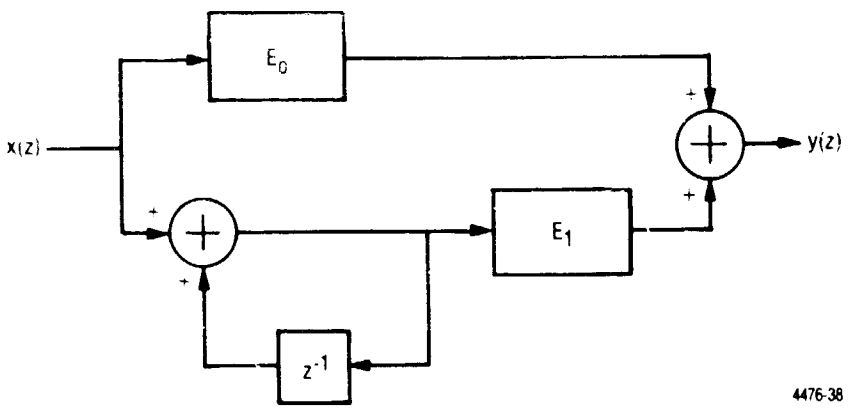
$$\hat{\theta}(z) = z^{-1}\hat{\theta}(z) + z^{-1}K\phi D(z) \qquad \text{(B-6)}$$

And the equivalent transfer function may be written as

$$\boxed{\frac{\hat{\theta}(z)}{D(z)} = \frac{K\phi}{z-1}} \qquad \text{(B-7)}$$

B.3 LOOP FILTER

The subcarrier tracking loop filter consists of the discrete-sample analog of a parallel combination of an ideal integrator and a direct (scaled) connection. The filter may be represented as



Then

$$F(z) = E_o + E_1 \left(\frac{z}{(z-1)} \right) \qquad \text{(B-8)}$$

APPENDIX C
HILBERT ACQUISITION AID ANALYSIS

In this appendix a noiseless analysis of the HAA is presented. Its purpose is to illustrate the effect on the control signal of dc offsets, quadrature phase errors, and quadrature amplitude imbalances. The functional block diagram of the HAA is shown in Figure C-1. Two cases will be analyzed – one without limiters and one with limiters. The input signal is

$$s(t) = \frac{A}{\sqrt{2}} m(t)\cos(\omega_{in}t + \theta_0) \tag{C-1}$$

where $m(t) = \pm 1$ and represents binary data. Thus, s(t) is a Binary Phase-Shift Keyed (BPSK) signal with a suppressed carrier at ω_{in} and a uniformly distributed phase θ_0. This signal is multiplied by a quadrature reference frequency at ω_r and resolved into in-phase and quadrature components,

$$I(t) = A_i m(t)\cos(\omega t + \phi) \tag{C-2}$$

$$Q(t) = A_q m(t)\sin(\omega t + \phi - \beta) \tag{C-3}$$

where

$$\omega = \omega_{in} - \omega_r \quad \text{and} \quad \phi = \theta_0 - \theta_r$$

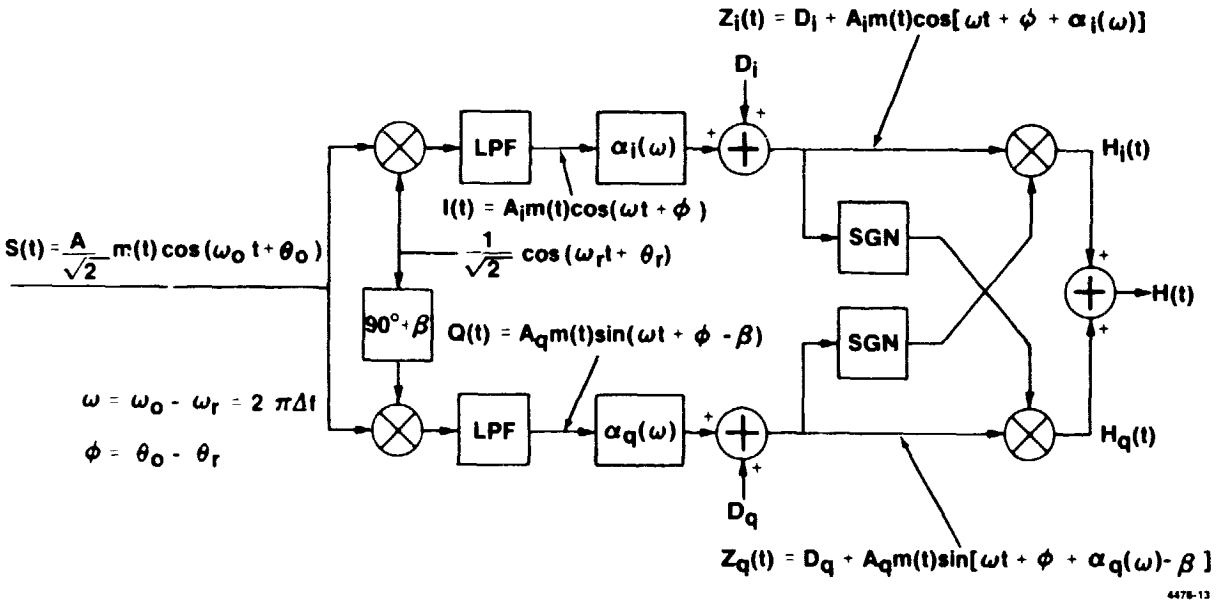


Figure C-1. Hilbert Acquisition Aid

It is assumed that the double frequency terms out of the multipliers are completely rejected and that an amplitude imbalance exists with A_i and A_q representing the resulting amplitudes. Also a quadrature phase error (β) is included in the analysis to permit an examination of its effects on the overall acquisition characteristic. These signals [I(t) and Q(t)] are then phase-shifted by $\alpha_i(\omega)$ and $\alpha_q(\omega)$, respectively. The resultant signals plus dc offsets D_i and D_q are

$$Z_i(t) = D_i + A_i m(t) \cos[\omega t + \phi + \alpha_i(\omega)] \tag{C-4}$$

$$Z_q(t) = D_q + A_q m(t) \sin[\omega t + \phi + \alpha_q(\omega) - \beta] \tag{C-5}$$

At this point the two analyses must diverge. The two cases considered in the following paragraphs are the true cross multiplication and the limiters designated by the SGN(·) function where,

$$\text{SGN}(X) = \begin{cases} 1 & X > 0 \\ +1 & X < 0 \end{cases}$$

C.1 HAA WITH CROSS-MULTIPLICATION

For this case, where the circuit is uncomplicated by the introduction of limiters, the analysis is straightforward. The output of the multipliers is,

$$H_i(t) = H_q(t) = Z_i(t)Z_q(t)$$

$$= D_i + A_i m(t) \cos[\omega t + \phi + \alpha_i(\omega)] \times \{D_q + A_q m(t) \sin[\omega t + \phi + \alpha_q(\omega) - \beta]\} \tag{C-6}$$

The summed characteristic is thus,

$$H(t) = 2\{D_i D_q + D_i A_q m(t) \sin[\theta + \alpha_q(\omega) - \beta] + D_q A_i m(t) \cos[\theta + \alpha_i(\omega)] + A_i A_q \cos[\theta + \alpha_i(\omega)] \sin[\theta + \alpha_q(\omega) - \beta]\} \tag{C-7}$$

where

$$\theta = \omega t + \phi.$$

Since the output of the summer is to be filtered by a narrowband lowpass, in effect a time average is taken. Thus, of more interest is the expectation of H(t) which would provide the dc forcing function for the frequency acquisition loop. This expectation is

$$\langle H(t) \rangle = 2D_i D_q + 2A_i A_q \langle \cos[\theta + \alpha_i(\omega)] \sin[\theta + \alpha_q(\omega) - \beta] \rangle \tag{C-8}$$

$$\langle H(t) \rangle = 2D_i D_q + 2A_i A_q \frac{1}{2\pi} \int_0^{2\pi} \cos[\theta - \alpha_i(\omega)] \sin[\theta + \alpha_q(\omega) - \beta] d\theta$$

$$= 2D_i D_q + A_i A_q \left\{ 0.5\theta \sin[\alpha_q(\omega) - \beta - \alpha_i(\omega)] - 0.25 \cos[2\theta + \alpha_i(\omega) - \beta] \right\} \Big|_0^{2\pi}$$

$$= 2D_i D_q - A_i A_q \sin(\psi + \beta) \tag{C-9}$$

where,

$$\psi = \alpha_i(\omega) - \alpha_q(\omega) \tag{C-10}$$

From (C-9) it can be seen that regardless of the data modulation [m(t)], reference phase (θ_r), or reference chopping, the dc offsets can have a significant effect. However, with periodic reference phase inversions or adequate data transition density, capacitive coupling is possible which will permit the offsets to be reduced to negligible levels. The quadrature phase error (β) can also be significant; however, with the proper attention β can be reduced to insignificant levels as well.

C.2 HAA WITH CROSS-CHANNEL LIMITERS

This case, with cross-channel limiters, has more practical significance than the previous case since the actual implementation will employ digitized values for $Z_i(t)$ and $Z_q(t)$. By using limiters a digital multiplier is not required and a large savings in power consumption is realized. However, the analysis is somewhat more complicated. For this case, the outputs of the multipliers become,

$$H_i(t) = \{D_i + A_i m(t)\cos[\omega t + \phi + \alpha_i(\omega)]\} \cdot \left(\text{SGN}\{D_q + A_q m(t)\sin[\omega t + \phi + \alpha_q(\omega) - \beta]\}\right) \quad \text{(C-11)}$$

$$H_q(t) = \{D_q + A_q m(t)\sin[\omega t + \phi + \alpha_q(\omega) - \beta]\} \cdot \left(\{D_i + A_i m(t)\cos[\omega t + \phi + \alpha_i(\omega)]\}\right)\text{SGN} \quad \text{(C-12)}$$

And taking the expectation of $H_i(t)$ and $H_q(t)$,

$$\begin{aligned}\langle H_i(t)\rangle &= \langle D_i \text{SGN}[D_q + A_q m(t)\sin(\theta_q - \beta)]\rangle \\ &+ \langle A_i m(t)\cos\theta_i \text{SGN}[D_q + A_q m(t)\sin(\theta_q - \beta)]\rangle \\ &= E_{i1} + E_{i2}\end{aligned} \quad \text{(C-13)}$$

and

$$\begin{aligned}\langle H_q(t)\rangle &= \langle D_q \text{SGN}[D_i + A_i m(t)\cos\theta_i]\rangle \\ &+ \langle A_q m(t)\sin(\theta_q - \beta)\text{SGN}[D_i + A_i m(t)\cos\theta_i]\rangle \\ &= E_{q1} + E_{q2}\end{aligned} \quad \text{(C-14)}$$

where

$$\theta_i = \omega t + \phi + \alpha_i(\omega) \text{ and } \theta_q = \omega t + \phi + \alpha_q(\omega) \quad \text{(C-15)}$$

Each of the terms E_{i1}, E_{i2}, E_{q1}, and E_{q2} must be evaluated individually. Therefore, proceeding in order

$$E_{i1} = \langle D_i \text{SGN}[D_q + A_q m(t)\sin(\theta_q - \beta)]\rangle \quad \text{(C-16)}$$

$$= \frac{D_i}{2\pi}\int_0^{2\pi} \text{SGN}[D_q + A_q m(t)\sin(\theta_q + \beta)]d\theta_q \quad \text{(C-17)}$$

$$= (2p-1)\frac{D_i}{2\pi}(\theta_q + \beta)\Big|_{\phi_q}^{\pi-\phi_q} - (2p-1)\frac{D_i}{2\pi}(\theta_q - \beta)\Big|_{\pi-\phi_q}^{2\pi+\phi_q}$$

where

$$\phi_q = \sin^{-1}\left(\frac{D_q}{A_q}\right),\ p = \text{Prob}\,[m(t) = 1] \text{ and } |D_q| < |A_q|$$

Then

$$E_{1i} = \begin{cases} -\dfrac{2D_i\theta_q}{\pi} \quad (2p-1) & |D_q| < |A_q| \\ D_i \text{SGN}(D_q) & |D_q| \geq |A_q| \end{cases} \tag{C-18}$$

Now solving for E_{12} in similar fashion,

$$E_{12} = \langle A_i m(t) \cos\theta_i \text{SGN}[D_q + A_q m(t) \sin(\theta_q - \beta)] \rangle \tag{C-19}$$

$$= \frac{1}{2\pi}\left[\int_{\phi_q}^{\pi-\phi_q} A_i \cos(\theta_q + \psi + \beta)d\theta_q - \int_{\pi-\phi_q}^{2\pi+\phi_q} A_i \cos(\theta_q + \psi + \beta)d\theta_q\right]$$

where

$$\psi = \theta_i - \theta_q = \alpha_i(\omega) - \alpha_q(\omega) \text{ and } |D_q| < |A_q|$$

$$E_{12} = \frac{A_i}{2\pi}\left[\int_{\theta_q+\psi+\beta}^{\pi-\theta_q+\psi+\beta} \cos\gamma d\gamma - \int_{\pi-\phi_q+\psi+\beta}^{2\pi+\phi_q+\psi+\beta} \cos\gamma d\gamma\right] \tag{C-20}$$

for $\gamma = \theta_q + \psi + \beta$

This results in

$$E_{12} = \begin{cases} -\dfrac{2A_i}{\pi}\cos\phi_q \sin(\psi+\beta) & |D_q| < |A_q| \\ 0 & |D_q| \geq |A_q| \end{cases} \tag{C-21}$$

Similarly for E_{11}

$$E_{11} = \langle D_i \text{SGN}[D + A\,m(t)\cos\theta_i]\rangle \tag{C-22}$$

$$= \frac{D_i}{2\pi}\int_0^{2\pi} \text{SGN}[D_i + A\,m(t)\cos\theta_i]\,d\theta_i \tag{C-23}$$

$$\frac{D_i}{2\pi}(2p-1)\theta\Big|_{\phi-\pi/2}^{\pi/2-\phi} - \frac{D_i}{2\pi}(2p-1)\theta\Big|_{\pi/2-\phi}^{3\pi/2+\phi}$$

where

$$\phi_i = \sin^{-1}(D_i/A_i) \text{ and } |D_i| < |A_i|$$

Then

$$E_{qi} = \begin{cases} -\frac{2D_q}{\pi}(2p-1)\,\phi_i & |D_i| < |A_i| \\ D_q \;\; \text{SGN}(D_i) & |D_i| \geq |A_i| \end{cases} \tag{C-24}$$

Continuing for E_{qi},

$$E_{qi} = \langle A_q m(t) \sin(\theta_q - \beta)\, \text{SGN}[D_i + A_i m(t) \cos\theta_i] \rangle \tag{C-25}$$

$$= \frac{A_q}{2\pi}\left[\int_{-\pi/2+\phi_i}^{\pi/2-\phi_i} \sin(\theta_i - \psi - \beta)\, d\theta_i - \int_{\pi/2-\phi_i}^{3\pi/2+\phi_i} \sin(\theta_i - \psi - \beta)\, d\theta_i\right]$$

where

$$|D_i| < |A_i|$$

Then

$$E_{qi} = \frac{A_q}{2\pi}\left[\int_{-\pi/2+\phi_i-\psi-\beta}^{\pi/2-\phi_i-\psi-\beta} \sin\gamma d\gamma - \int_{\pi/2-\phi_i-\psi-\beta}^{3\pi/2+\phi_i-\psi-\beta} \sin\gamma d\gamma\right] \tag{C-26}$$

After considerable trigonometric manipulation this results in

$$E_{qi} = \begin{cases} -\frac{2A_q}{\pi}\cos\phi_i \sin(\psi+\beta) & |D_i| < |A_i| \\ 0 & |D_i| \geq |A_i| \end{cases} \tag{C-27}$$

Finally, combining terms, the HAA control voltage becomes,

$$\langle H(t) \rangle = -\frac{2D_i\,\phi_q}{\pi}(2p-1) - \frac{2D_q\,\phi_i}{\pi}(2p-1) - \frac{2A_i}{\pi}\cos\phi_q \sin(\psi+\beta) - \frac{2A_q}{\pi}\cos\phi_i \sin(\psi+\beta) \tag{C-28}$$

when $\begin{matrix} |D_i| < |A_i| \\ |D_q| < |A_q| \end{matrix}$

$$\langle H(t) \rangle = D_q \text{SGN}(D_i) + D_i \text{SGN}(D_q) \text{ when } \begin{matrix} |D_i| \geq |A_i| \\ |D_q| \geq |A_q| \end{matrix} \tag{C-29}$$

When the dc offsets are negligible, the control voltage becomes,

$$\langle H(t) \rangle = -\frac{2}{\pi}(A_q + A_i)\sin(\psi+\beta) \tag{C-30}$$

These results are essentially the same as those for the crossmultiplication case analyzed in paragraph C.1 except that now transition probabilities begin to play a role. The dc offsets and quadrature imbalances must still be controlled.

Actual expected degradations are determined in paragraph C.3 and indicate that the effect on the acquisition characteristic is negligible.

C.3 HAA ERROR ANALYSIS

The only significant error contributions to the HAA output characteristic are dc offsets and quadrature error. Assuming an adequate transition density, or periodic software-controlled phase inversions, ac coupling is permissable. This reduces the dc offset consideration to two components, the ADC and the sample-and-hold (S/H) amplifier. The ADC design under consideration would have a worst case offset of ±1/2 LSB. This results in an offset of ±20 mV for an 8-bit ADC with ±5 volt saturation levels. The overall channel offset is adjusted with a worst case settability of ±5 mV through the ADC, leaving only the S/H amplifier drift components. The magnitude of this component is typically 20 μV/°C. Thus, the offset voltage over the full temperature range −55°C to +125°C is

$$V_{os} = \pm V_{ADC} \pm V_{set} \pm V_D \qquad \text{(C-31)}$$

where V_{ADC} is the ADC offset voltage, V_{set} is the offset settability and V_D is the worst case drift offset. Thus from the above:

$$V_{os} = \pm 20\ \text{mV} \pm 5\ \text{mV} \begin{matrix} +\ 2\ \text{mV} \\ -\ 1.6\ \text{mV} \end{matrix}$$

or, the worst case dc offset is 27 mV.

For the quadrature error, the 90-degree hybrid and differential circuit path lengths at the IF frequency are the major error contributors. For the 90-degree hybrid, the quadrature error can be held to ±1 degree absolute at the IF frequency with an additional error of ±0.5 degree due to the ±100 kHz uncertainty about an IF frequency of 20 MHz. The differential path lengths can be held to ±0.1 inch for an error of ±0.0125 nanosecond, or 0.1 degree. If the differential path lengths are used to adjust the total absolute error through the hybrid as well, then the total absolute error is reduced to ±0.1 and ±0.5 degree due to the frequency offset. Thus, the total quadrature phase error can be held under one degree.

To evaluate the effects of these errors the signal level must be determined. For a peak signal amplitude, A, white noise power spectral density (N_o) and input noise bandwidth B_n the ADC saturation probability can be derived as follows:

The desired probability of saturation is for an instantaneous signal plus noise sample (X_i) exceeding the ADC saturation voltage (V_s) or

$$\text{Prob}\,(|X| > V_s) = 1 - \text{Prob}\,(-V_s < X < V_s) = P$$

The desired probability of saturation is for an instantaneous signal plus noise sample (X_i) exceeding the ADC saturation voltage (V_s) or

$$\text{Prob}\{|X_i| > V_s\} = 1 - \text{Prob}\{-V_s < X_i < V_s\} = P_s$$

$$P_s \leq 1 - \frac{1}{\sigma\sqrt{2\pi}} \int_{-V_s}^{V_s} \exp\left[-(X - A)^2/2\sigma^2\right] dx$$

$$\leq 1 - P\left(\frac{V_s - A}{\sigma}\right) + P\left(\frac{-V_s - A}{\sigma}\right) \approx Q\left(\frac{V_s - A}{\sigma}\right) \quad \text{(C-32)}$$

where[1]

$$P(X) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{X} \exp\left[-t^2/2\right] dt \quad \text{(C-33)}$$

$$Q(X) = \frac{1}{\sqrt{2\pi}} \int_{X}^{\infty} \exp\left[-t^2/2\right] dt \quad \text{(C-34)}$$

then,

$$\frac{V_s - A}{\sigma} \geq Q^{-1}(P_s)$$

and,

$$\text{with } \sigma = \frac{A}{\sqrt{2\rho}} \text{ volts where } \rho = \frac{S}{N_0 B_n}$$

$$A \leq \frac{V_s - Q^{-1}(P_s)}{1 + \frac{1}{\sqrt{2\rho}}} \quad \text{(C-35)}$$

Now, for $V_s = 5$ volts, $P_s = 0.1$, $B_n = 10^5$ Hz, and $S/N_0 = 24.5$ dB Hz

$$A \leq \frac{5 - 1.28155}{1 + 13.32} = 0.26 \text{ volt} \quad \text{(C-36)}$$

Thus, the maximum signal level at the design S/N_0 should be 260 mV, and the ratio of worst case dc offset to peak signal level is approximately 0.1. This signal level is adequate, as determined by past experience, to prevent significant losses due to ADC quantization for an 8-bit ADC with $V_s = 5$ volts.

[1]Abramowitz and Stegun, "Handbook of Mathematical Functions," U.S. Department of Commerce, National Bureau of Standards Applied Mathematics Series, 1966, p. 931.

In the following plots this ratio of DC offset to peak signal level and the quadrature error of one degree is shown. Because the results in paragraphs C.1 and C.2 were similar, only the results of paragraph C.2 are presented here since these results apply directly to the preferred implementation. First in Figure C-2, the phase shifts $\alpha_i(\omega)$, $\alpha_q(\omega)$, and $\psi(\omega)$ are plotted for the actual hardware implementation. As shown $\psi(\omega)$ approximates the Hilbert transform with an equi-ripple phase differential between frequencies F_1 = 1000 Hz and F_2 = 50 kHz.

In Figure C-3 the control signal $< H(t) >$ is plotted with no degradations. In Figure C-4 the control signal with the combined worst case degradations is shown. A worst case for the probability of a one being transmitted, P, is assumed as P = 0.6. As seen in Figure C-4, the zero crossing offset of the control voltage characteristic occurs at 6 Hz. This error is not significant for the initial acquisition phase during which the HAA will be active.

A second case where F_1 = 100 Hz and F_2 = 50 kHz is illustrated with a similar set of plots in Figures C-5 through C-7. This case has a much improved acquisition characteristic about $\omega = 0$ at the expense of a larger but insignificant amount of ripple for frequencies between F_1 and F_2. For cases where the frequency uncertainty is less than 50 kHz this characteristic should provide superior performance. As seen in Figure C-7, for the same set of worst-case degradations, the effect on the zero crossing offset is much reduced, with the zero crossing occurring within one Hertz of zero frequency.

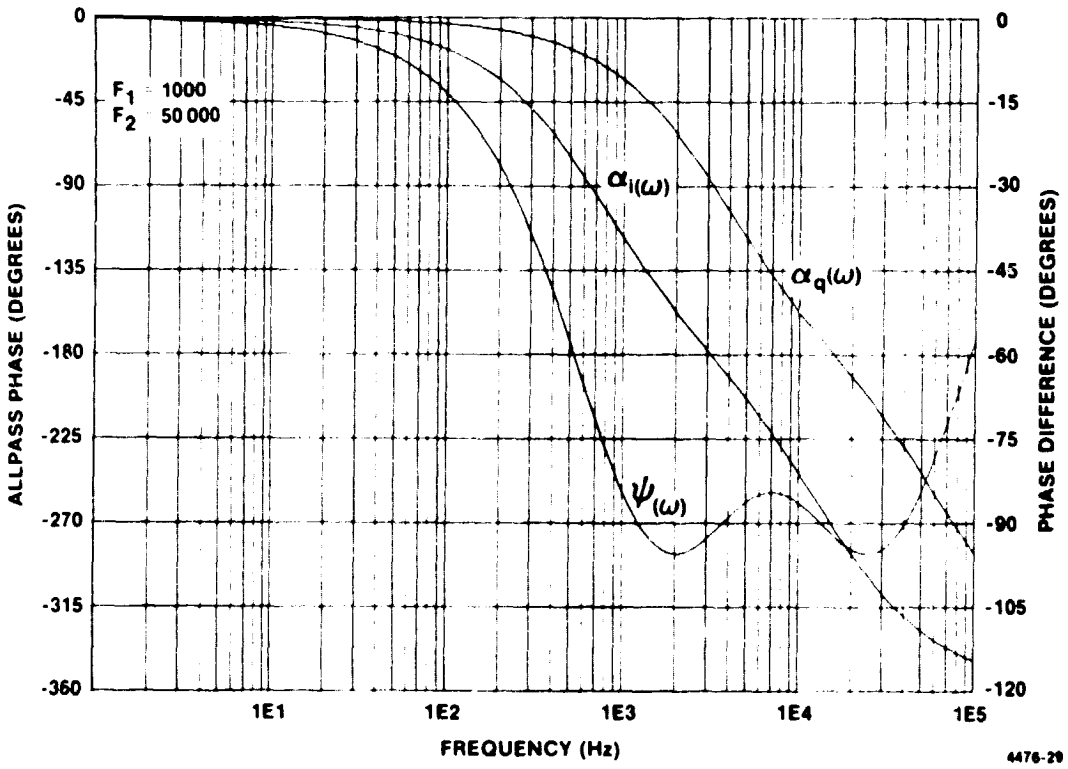


Figure C-2. HAA Phase Angle Versus Frequency

ORIGINAL PAGE IS
OF POOR QUALITY

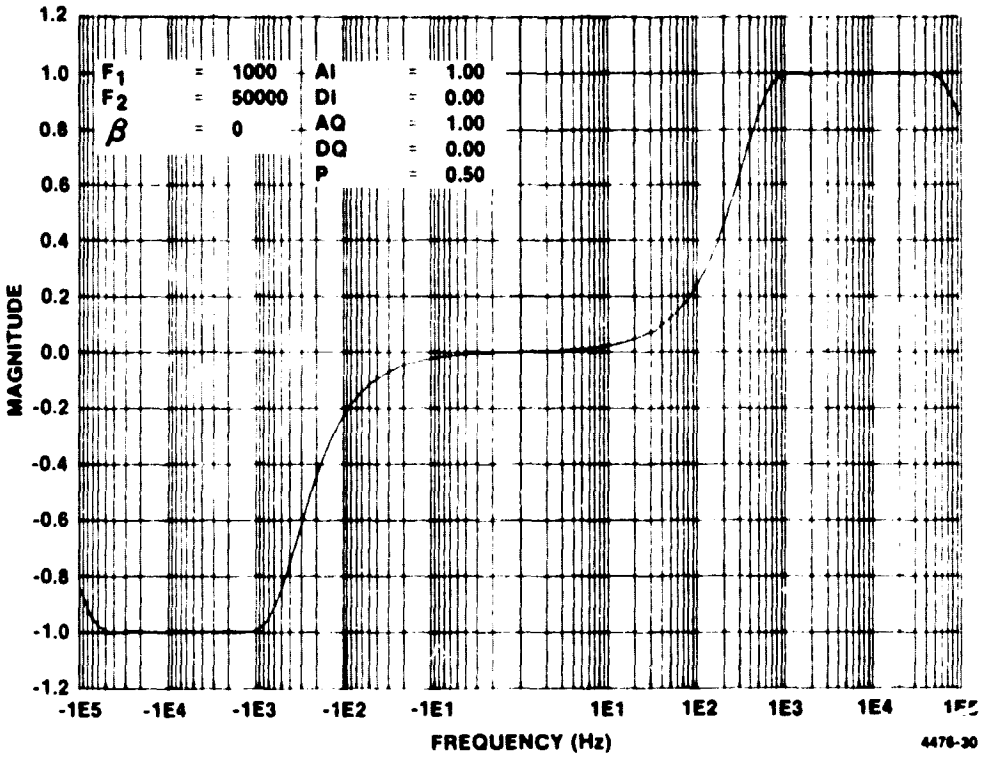


Figure C-3. HAA Voltage Versus Frequency

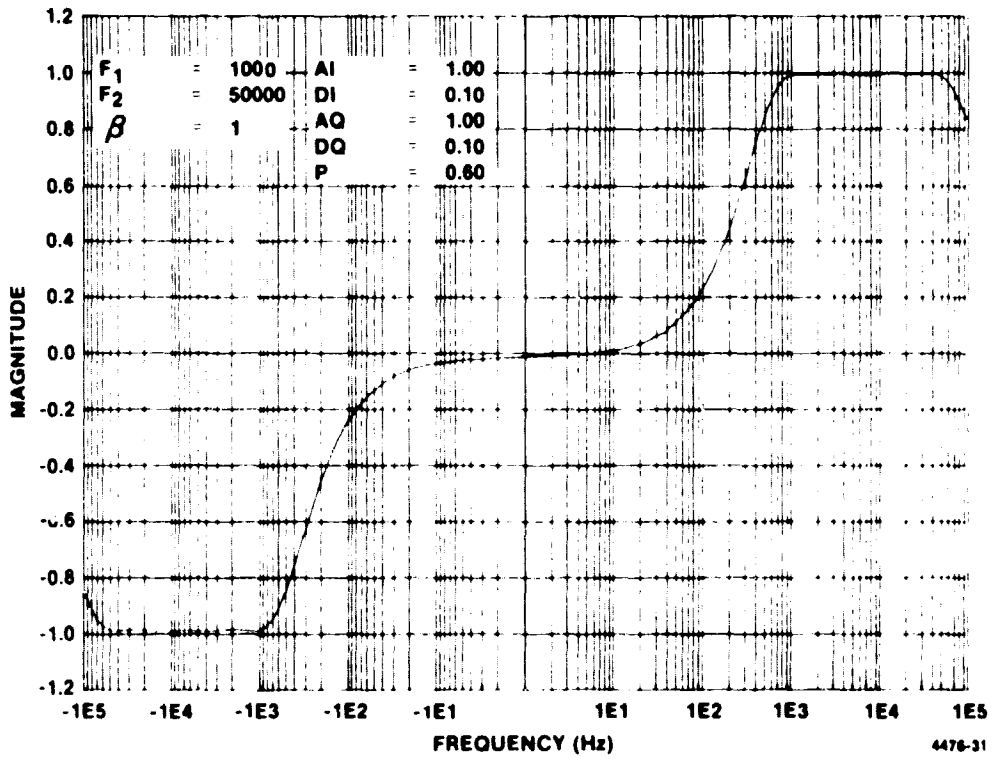


Figure C-4. HAA Voltage Versus Frequency

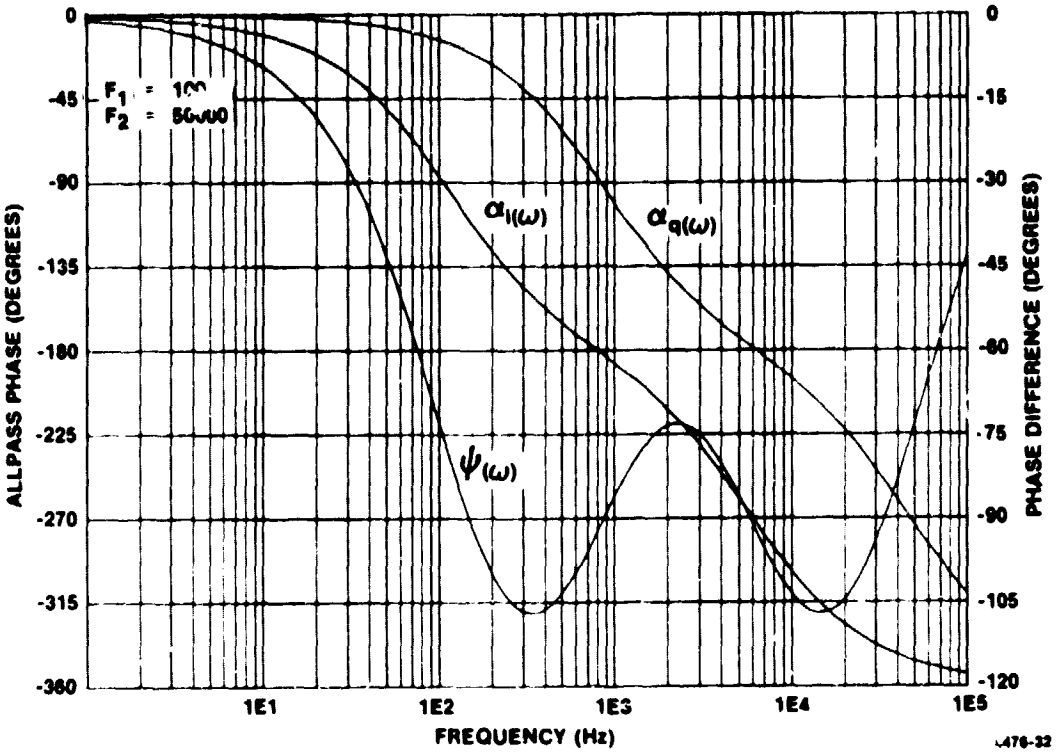


Figure C-5. HAA Phase Angle Versus Frequency

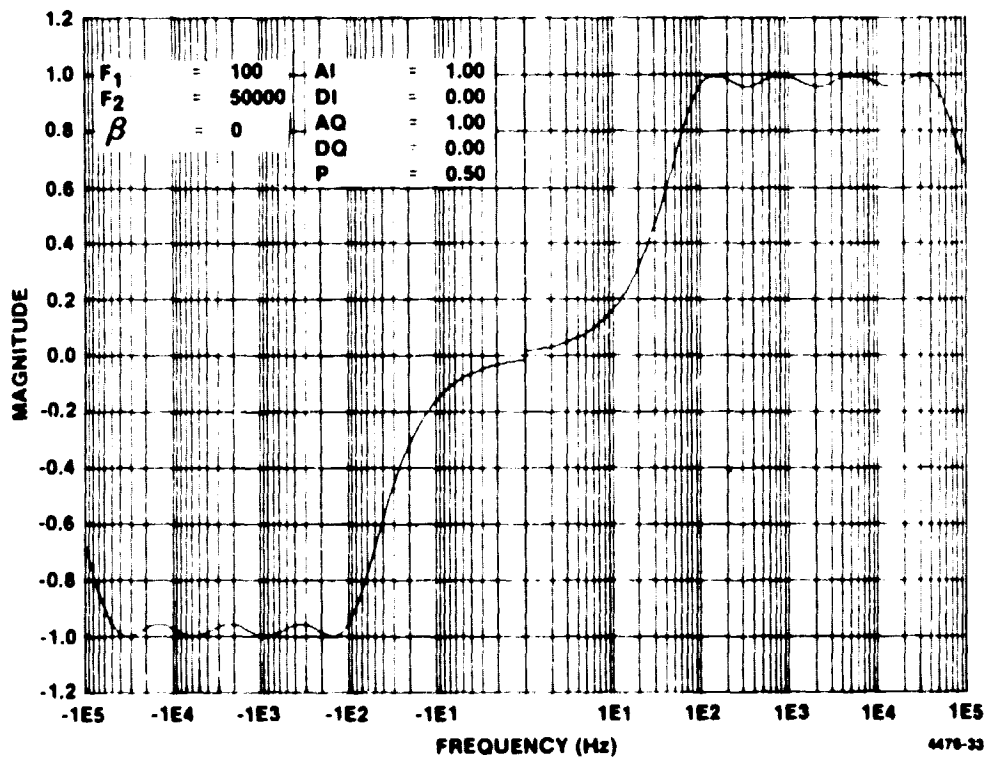


Figure C-6. HAA Voltage Versus Frequency

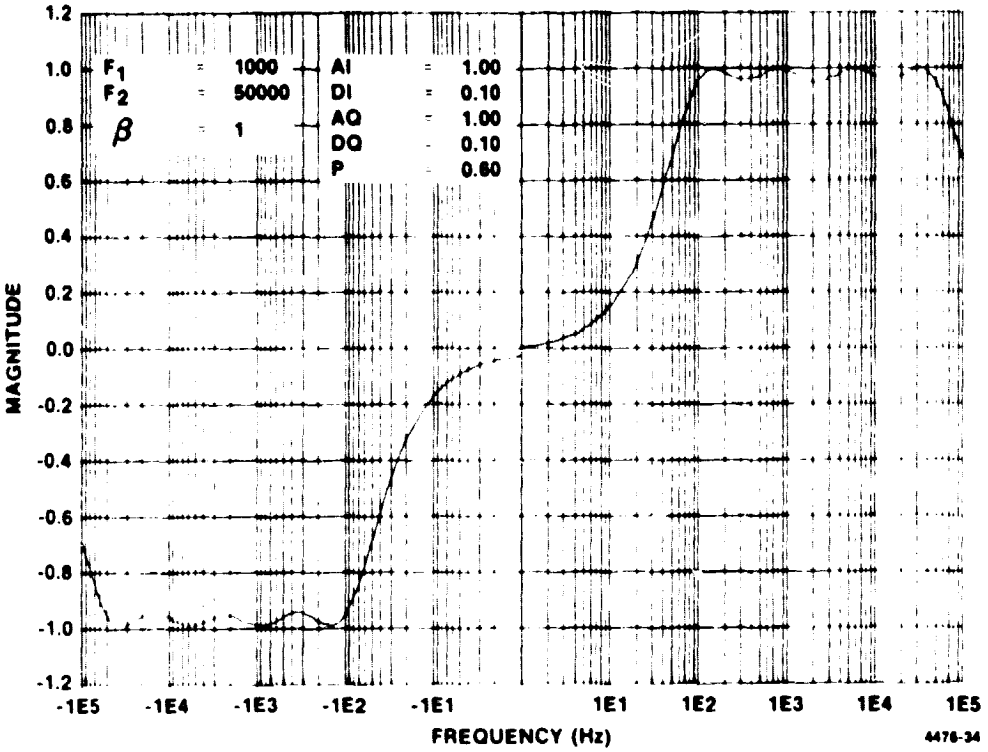


Figure C-7. HAA Voltage Versus Frequency

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX D
NASA STANDARD CDU MICROPROCESSOR: ARCHITECTURE, ASSEMBLER, SIMULATOR

This appendix describes the NASA Standard CDU Microprocessor, a microprocessor designed specifically for the implementation of the command detection algorithms. It also describes the CDU Assembler, a program written to make possible the programming of the CDU Microprocessor. The CDU Assembler is a FORTRAN program running on the Engineering Computer at Motorola GED.

D.1 ARCHITECTURE

D.1.1 Overview

- Maximum clock rate: 1.2 MHz at V_{DD} = +12 volts
- ROM: 1024 16-bit words
- RAM: 64 16-bit words
- Two 16-bit ALUs, each with two 16-bit arithmetic registers, one 5-bit shift count register, and 16 operations
- Eight program flags, three input flags, four condition code flags
- One 5-bit offset register
- Seven I/O devices
- Three instruction types:
  - Moves - 35
  - Branches - 32
  - ALU Instructions - 13

D.1.2 Detailed Description

D.1.2.1 ROM

The ROM is used for program and read-only data storage. Its maximum extent is 1024 words, addressable as 0, . . . , 1023. Words are 16-bits long.

D.1.2.2 RAM

The RAM is used for data storage. Its maximum extent is 64 words, addressable as 0, . . . , 63. The present implementation contains 32 words, addressable as 0, . . . , 31.

D.1.2.3 ALUs

The two ALUs, ALU1 and ALU2, perform 16-bit two's-complement arithmetic using the values in their two 16-bit arithmetic registers [(A1, B1) and (A2, B2)] and their one 5-bit shift count register (SCR1 and SCR2). Of the 16 possible operations, 13 are used:

COMMAND CODE	MEANING	FLAGS AFFECTED (ALU1 only)
0000	No-op	None
0001	$B_n = A_n + B_n$	N, Z, V, C
0010	$B_n = B_n + 1$	N, Z, V, C
0011	Undefined	None
0100	$(B_n, A_n) := (B_n, A_n)$ ASR[1] SCR_n	N, Z
0101	$B_n = B_n$ ASR[1] SCR_n	Z
0110	$B_n = -B_n$	N, V, C
0111	$B_n = A_n - B_n$	N, Z, V, C
1000	$B_n = \lvert B_n \rvert$	N, V, C
1001	Undefined	None
1010	Load SCR_n	None
1011	Undefined	None
1100	Store B_n	None
1101	Store A_n	None
1110	Load A_n	None
1111	Load B_n	N, Z

[1] Arithmetic shift right: Must have $1 \leq SCR_n \leq 31$.

D.1.2.4 FLAGS

Flags provide single-bit data storage. The eight program flags P0, . . ., P7 may be set to 0 or 1 by the program. The three input flags I1, I2, I3 are set to 0 or 1 under external control. The four condition code flags N, Z, V, C are set to 0 or 1 based on the value in B1 as follows:

FLAG	MEANING IF 0	MEANING IF 1
N	$B1 \geq 0$	$B1 < 0$
Z	$B1 \neq 0$	$B1 = 0$
V	B1 did not overflow	B1 overflowed
C	No carry out of B1	Carry out of B1

All 15 flags may be tested by the program in conditional branches. In addition, the input flags and the condition code flags may be read by move instructions.

D.1.2.5 OFFSET REGISTER

The offset register, OR, is a 5-bit register capable of receiving a number between 0 and 31 inclusive. Its contents are used by the relative branch instruction.

D.1.2.6 DEVICES

A maximum of seven devices DEV1, . . ., DEV7 are available for I/O.

- DEV1, Command Input

Microprocessor Input Format

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B	B	B	B	B	B	B	B	B	A	A	A	A	A	A	A

Bits 0 through 7 represent the code corresponding to the command input and bits 8 through 15 are the replicated MSB of the command input.

- DEV2, Data and Clock Output

Microprocessor Output Format

(1)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

Midbit.

(2)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	D

D = 0: end of bit, data = 1.

D = 1: end of bit, data = −1.

(3)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	L_1	L_2

$L_1 = 0$: no lock 1.

$L_1 = 1$: lock 1.

$L_2 = 0$: no lock 2.

$L_2 = 1$: lock 2.

- DEV3, Telemetry

Microprocessor Output Format

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T	T	T	T	T	T	T	T	T	T	T	T	T	T	T	T

T is the telemetry output word.

- DEV4, Automatic Gain Control

Microprocessor Output Format

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	A	A	A	A	A	A	A	A	A	A

A is the AGC control word.

D.1.2.7 INSTRUCTION TYPES

The three instruction types are ALU commands, moves, and branches. Their formats are summarized in this table:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
0	0	0	0	0	0	L	L	L	L	L	L	L	L	L	L	Unconditional branch.
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	Relative branch.
B	B	B	B	B	M	L	L	L	L	L	L	L	L	L	L	Conditional branch.
1	1	1	0	0	1	0	0	0	1	1	A	X	X	X	X	ALU command.
1	1	D	D	D	D	D	D	D	S	S	S	S	S	S	S	Move.

- ALU Commands

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	1	0	0	1	0	0	0	1	1	A	X	X	X	X

Execution of this instruction causes ALU A + 1 to perform the operation with command code X (paragraph D.1.2.3). X refers to one of the eight arithmetic operations: no-op, add, increment, double arithmetic shift right, single arithmetic shift right, negate, subtract, and absolute value.

- Moves

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	1	D	D	D	D	D	D	D	S	S	S	S	S	S	S

Execution of this instruction causes the value at location S (the "source" location) to be moved to location D (the "destination" location).

The allowable formats for the source and destination fields are:

D D D D D D D	S S S S S S S	MEANING
0 R R R R R R	0 R R R R R R	RAM address R
1 0 0 0 P P P		Program flag P (0-7 for P0-P7, respectively)
	1 0 0 1 F F F	Input or condition code flag F (1-3 for I1-I3, 4,5,6,7 for N,Z,V,C, respectively)
	1 0 1 X X X X	Constant X
	1 1 0 0 0 0 0	Following word in ROM (moved next instruction cycle)
1 1 0 0 D D D	1 1 0 0 D D D	Device D (1-7 for DEV1-DEV7, respectively)
1 1 1 0 0 0 0		Offset register
1 1 A Z Z Z Z	1 1 A Y Y Y Y	ALU A + 1 register denoted by Y and Z

Where Y and Z have the following meanings:

Z Z Z Z	Register	Y Y Y Y	Register
1 1 1 0	A	1 1 0 1	A
1 1 1 1	B	1 1 0 0	B
1 0 1 0	SCR		

Remarks:

(1) Note from paragraph D.1.2.3 that when the destination D is 1101111, the N and Z condition code flags may be affected.

(2) These types of moves are not allowed:

(i) RAM to RAM,

(ii) device to device,

(iii) ALU 1 register to ALU 1 register, and

(iv) ALU 2 register to ALU 2 register.

(3) Inputs from and outputs to a device are accomplished by specifying the device as the source or the destination, respectively. For example, data transfer from RAM location 3 to Device 2 is accomplished by executing the instruction.

1 1 1 1 0 0 0 1 0 0 0 0 0 0 1 1 .

- Branches

The various types of branch instructions are described below:

Conditional Absolute

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B	B	B	B	B	M	L	L	L	L	L	L	L	L	L	L

Execution of this instruction causes a branch to program word L in the ROM if the indicated flag has the value M. The flag is indicated as follows:

B B B B B	Flag	B B B B B	Flag
0 1 0 0 0	P0	0 1 1 0 0	I1
1 0 0 0 0	P1	1 0 1 0 0	I2
0 1 0 0 1	P2	0 1 1 0 1	I3
1 0 0 0 1	P3	1 0 1 0 1	N
0 1 0 1 0	P4	0 1 1 1 0	Z
1 0 0 1 0	P5	1 0 1 1 0	V
0 1 0 1 1	P6	0 1 1 1 1	C
1 0 0 1 1	P7		

An extra instruction time for flag settling must be allowed between instructions that set flags and conditional branches that access those flags.

Unconditional Absolute

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	L	L	L	L	L	L	L	L	L	L

Execution of this instruction causes a branch to program word L in the ROM.

Unconditional Relative

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0

Execution of this instruction causes a branch to the program word in the ROM whose address is the sum of the address of the instruction following this one and the contents of the offset register.

An extra instruction time for offset register settling must be allowed between instructions that load the offset register and relative branches.

D.2 ASSEMBLER

D.2.1 Operation

This section defines an assembly language used to write programs for the NASA Standard CDU Microprocessor. The assembler accepts a card deck or a source file from the disc containing a microprocessor program written in that

language and produces: a listing showing in hexadecimal the numbers that would constitute the memory contents if the program were loaded; a paper or magnetic tape which may be used to load the breadboard memory; a card deck which may be used to load the flight ROM. Output of the paper tape, magnetic tape or card deck may be suppressed if desired.

The assembler is run by submitting the following cards:

```
!JOB . . .
!ASSIGN F:104, (DEVICE, NO), (OUT) (only if paper tape not desired)
!ASSIGN F:106, (DEVICE, NO), (OUT) (only if card deck not desired)
!RUN (LMN, HASM)
!DATA
. . . (program) . . .
```

The program is keypunched one statement per card. Any statement other than an assembler directive statement (paragraph D.2.2.4) may begin with a label identifying that statement as a possible branch *target*. Labels, if present, must begin in column 1. They are the only alphanumeric strings which may begin in column 1. Except for this condition, a statement may be positioned anywhere on the card, with at least one space between any two adjacent alphanumeric strings.

Labels, RAM locations, and user-defined constants are strings consisting of from 1 to 8 alphanumeric characters, the first of which must be a letter. Numbers consist of an optional sign followed by decimal or hex digits, with the number assumed hex if its first digit is 0 and decimal otherwise.

Any card with an asterisk in column 1 is treated as a comment, as is any text following the statement on a card. Each statement has one of four forms: no-operand, single-operand, assignment, and assembler directive.

D.2.2 Statement Forms

D.2.2.1 NO-OPERAND STATEMENTS

No-operand statements define ALU commands and the relative branch. They consist of a single mnemonic which is one of the following:

Mnemonic	Meaning	Mnemonic	Meaning
INC1	ALU 1 increment	INC2	ALU 2 increment
ABS1	ALU 1 absolute value	ABS2	ALU 2 absolute value
NEG1	ALU 1 negate	NEG2	ALU 2 negate
SR1	ALU 1 shift right	SR2	ALU 2 shift right
DSR1	ALU 1 double shift right	DSR2	ALU 2 double shift right
ADD1	ALU 1 add	ADD2	ALU 2 add
SUB1	ALU 1 subtract	SUB2	ALU 2 subtract
NOP	no operation	BR	relative branch

D.2.2.2 SINGLE-OPERAND STATEMENTS

Single-operand statements define all other branches. They consist of one of the following mnemonics followed by a label identifying the statement at which execution is to continue:

Mnemonic	Branch Condition	Mnemonic	Branch Condition
B	Unconditional	BP3	Program flag 3 set
BN	Negative	BNP3	Program flag 3 clear
BNN	Not negative	BP4	Program flag 4 set
BZ	Zero	BNP4	Program flag 4 clear
BNZ	Not zero	BP5	Program flag 5 set
BV	Overflow	BNP5	Program flag 5 clear
BNV	No overflow	BP6	Program flag 6 set
BC	Carry	BNP6	Program flag 6 clear
BNC	No carry	BP7	Program flag 7 set
BP0	Program flag 0 set	BNP7	Program flag 7 clear
BNP0	Program flag 0 clear	BI1	Input flag 1 set
BP1	Program flag 1 set	BNI1	Input flag 1 clear
BNP1	Program flag 1 clear	BI2	Input flag 2 set
BP2	Program flag 2 set	BNI2	Input flag 2 clear
BNP2	Program flag 2 clear	BI3	Input flag 3 set
		BNI3	Input flag 3 clear

D.2.2.3 ASSIGNMENT STATEMENTS

Assignment statements define moves. They consist of a destination operand and a source operand separated by the symbol "=" (becomes), where the operand on the left of the "=" symbol is the destination. An operand can be any of the following:

1. A RAM location, signified by an otherwise-undefined alphanumeric string.
2. A number.
3. A user-defined constant or user-defined name (paragraph D.2.2.4).
4. Any of the following mnemonics, which identify computer registers, flags, and peripherals:

Mnemonic	Meaning	Mnemonic	Meaning
P0	Program flag 0	OR	Offset register
P1	Program flag 1	DEV1	Device 1
P2	Program flag 2	DEV2	Device 2
P3	Program flag 3	DEV3	Device 3
P4	Program flag 4	DEV4	Device 4
P5	Program flag 5	DEV5	Device 5
P6	Program flag 6	DEV6	Device 6
P7	Program flag 7	DEV7	Device 7
I1	Input flag 1	A1	ALU 1 A register
I2	Input flag 2	B1	ALU 1 B register
I3	Input flag 3	SCR1	ALU 1 shift count register
N	Negative flag		
Z	Zero flag	A2	ALU 2 A register
V	Overflow flag	B2	ALU 2 B register
C	Carry flag	SCR2	ALU 2 shift count register

D.2.2.4 ASSEMBLER DIRECTIVE STATEMENTS

Assembler directive statements are END, PAGE, and equates:

- END

  The END statement consists of the single mnemonic END. It signifies the end of the program to the assembler. It is optional.

- PAGE

  The PAGE statement consists of the mnemonic PAGE optionally followed by a nonblank character string. It causes the assembler to start a new page at this point in the assembly listing. If a nonblank character string follows PAGE, it becomes the new page heading.

- Equates

  Equates consist of two operands separated by the sumbol ": :" (equals). The leftmost operand must be a previously-undefined alphanumeric character string. The rightmost operand may be any described in paragraph D.2.2.3. An equate defines the leftmost operand to the assembler with the value given by the rightmost symbol. For example, the statements

```
K0 : : 0
ADC : : DEV1
```

define the user-defined constant K0 to be equivalent to the number 0 and the user-defined name ADC to refer to device 1.

D.2.3 Output

D.2.3.1 ERROR MESSAGES

FIRST SYMBOL ILLEGAL.

- Before listing

  The first symbol on the following card (after the label, if one is present) is neither a mnemonic defined in D.2.2, an alphanumeric string referring to a RAM location, a previously-undefined alphanumeric string, nor a user-defined name.

- During listing

  Either

  The first symbol on the following card (after the label, if one is present) is neither a mnemonic defined in paragraph D.2.2, an alphanumeric string referring to a RAM location, a user-defined constant, nor a used-defined name.

  or

  The first symbol on the following card (after the label, if one is present) is a user-defined constant but the second symbol is not " : :".

SECOND SYMBOL ILLEGAL.

- Before listing

  Either

  The first symbol on the following card (after the label, if one is present) is either a mnemonic defined in paragraph D.2.2.3, an alphanumberic string referring to a RAM location, or a user-defined name, but the second symbol is not "=".

  or

  The first symbol on the following card (after the label, if one is present) is a previously-undefined alphanumeric string, but the second symbol is neither " : :" nor "=".

- During listing

  The first symbol on the following card (after the label, if one is present) is either a mnemonic defined in paragraph D.2.2.3, an alphanumeric string referring to a RAM location, or a user-defined name, but the second symbol is neither ": :" nor "=".

THIRD SYMBOL ILLEGAL IN EQUATE.

The first symbol on the following card (after the label, if one is present) is a previously-undefined alphanumeric string and the second symbol is ": :", but the third symbol is not an operand described in paragraph D.2.2.3.

THIRD SYMBOL ILLEGAL IN MOVE.

- Before listing:

  The first symbol on the following card (after the label, if one is present) is either a mnemonic defined in paragraph D.2.2.3, an alphanumeric string referring to a RAM location, a user-defined name, or a previously-undefined alphanumeric string, and the second symbol is "=", but the third symbol is neither an operand described in paragraph D.2.2.3 nor a previously-undefined alphanumeric string.

- During listing:

  The first symbol on the following card (after the label, if one is present) is either a mnemonic defined in paragraph D.2.2.3, an alphanumeric string referring to a RAM location, or a user-defined name, and the second symbol is "=", but the third symbol is not an operand described in paragraph D.2.2.3.

RAM OVERFLOW.

The microprocessor program needs more than 64 words or RAM.

ID TABLE OVERFLOW.

The microprocessor program uses more than 250 identifiers.

LABEL ALREADY DEFINED.

The symbol used as a label on the following card has been defined previously.

COLON MISUSED.

The following card contains a colon not immediately followed by another colon.

UNDEFINED LABEL.

The following card contains a single-operand statement (paragraph D.2.2.2). The symbol following the branch mnemonic is not a label.

APPENDIX E
SNR CALIBRATION OF THE HYBRID RECEIVER

In order to make the Hybrid Receiver acquisition and bit error rate tests meaningful it is necessary to determine the signal-to-noise ratio at which the tests were performed. The SNR measurements are taken using the configuration shown in Figure E-1. The calibrator filter shown is a two-pole, low-pass filter with a dc gain of 100 and an empirically determined single-sided noise bandwidth of 8585 Hz. The measurements are performed as follows. With the data turned off, the NCO frequency is offset by 1 kHz and the calibrator filter is connected to the I-channel output of the RF section. A measurement of the signal plus noise power in a 6 Hz bandwidth is taken using an HP302A Frequency Selective Voltmeter tuned to 1 kHz. A measurement is then made of the total signal plus noise power at the output of the filter using an HP3400A True RMS Voltmeter. The signal-to-noise ratio can then be calculated using the equations below:

$$N_0 = \frac{V_n^2 - V_m^2}{B_n - B_m} \tag{E-1}$$

$$S = V_m^2 - N_0 B_m \tag{E-2}$$

$$\frac{ST}{N_0} = \frac{S}{N_0}\left(\frac{1}{R}\right) \tag{E-3}$$

where

V_n = Total signal + noise (measured using HP3400A)
V_m = Signal + noise in 6 Hz bandwidth (measured using HP302A)
B_n = Calibrator filter noise bandwidth (8585 Hz)
B_m = Bandwidth of HP302A (6 Hz)
N_0 = Noise spectral density
S = Signal power
R = Data rate

Measurements are taken for two settings of the variable attenuator and then the process is repeated with the calibrator filter connected to the Q-channel output. The results are compared to verify that the measurements taken are accurate.

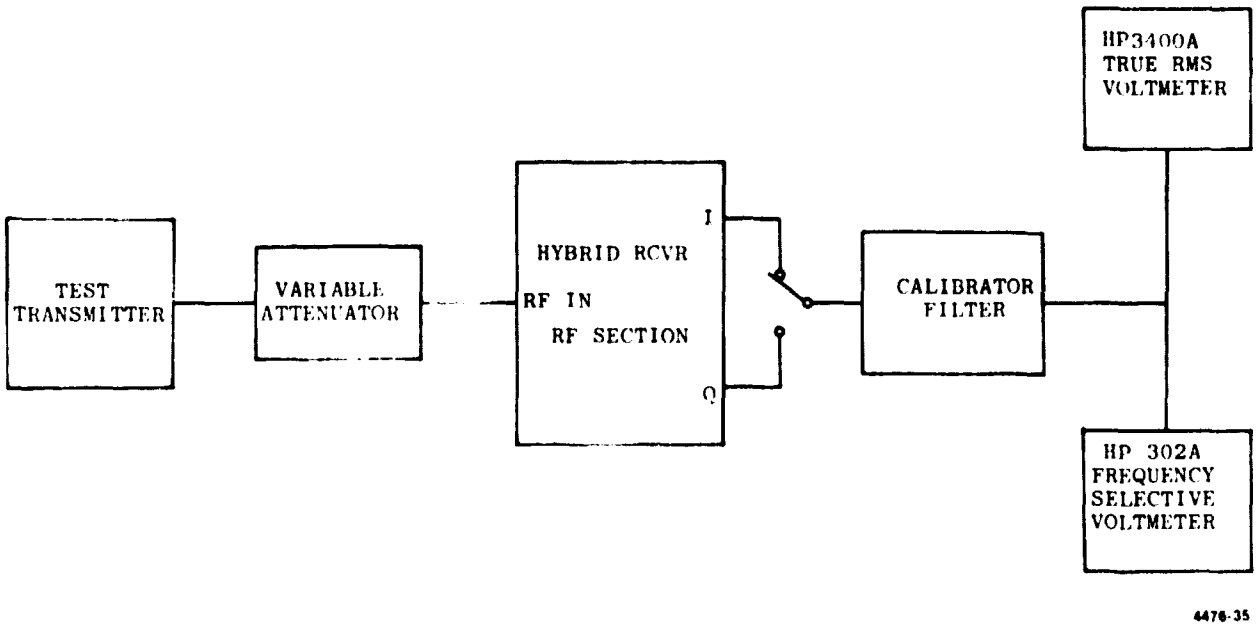


Figure E-1. S/N_0 Calibration Arrangement

ORIGINAL PAGE IS
OF POOR QUALITY

APPENDIX F

HYBRID RECEIVER PARTS LIST AND QUALIFICATION LEVEL

The preliminary parts list and qualification level of parts used in the Hybrid Receiver Digital Subsystem is given in Table F-1. All of the custom LSI circuits built by Motorola will be screened to the MIL-883 Class A- level. This means the custom devices will be built and screened to the MIL-883 Class A level with certain of the Class A tests omitted. Where possible parts that are qualified to MIL-M38510 will be used in the Hybrid Receiver. Purchased parts that are not available as MIL-M38510 will be screened to a vendor Hi-Rel MIL-883 Class B level or better. A summary of the parts used in the RF section of the Hybrid Receiver is given in Table F-2.

Table F-1. Parts List - Digital Subsystem

Quantity	Part Number	Description	Procurement Level
1	—	Program Sequencer[1]	MIL-883, Class A
1	—	Instruction Decoder[1]	MIL-883, Class A
1	—	Data Bus Interface[1]	MIL-883, Class A
4	—	ALU[1]	MIL-883, Class A
1	—	Data Router[1]	MIL-883, Class A
1	—	Dedicated Accumulator[1]	MIL-883, Class A
1	—	Numerically Controlled Oscillator[1]	MIL-883, Class A
1	—	Analog-to-Digital Converter[1]	MIL-883, Class A
3	CDP1834D	1024 x 8 ROM	MIL-883, Class B
2	CDP1823SD	128 x 8	MIL-883, Class B
1	DM75S28	1024 x 8 ROM	MIL-883, Class B
4	SN54LS04	Hex Inverter	38510, Class B
2	SN54LS163	Binary Counter	MIL-883, Class B
1	SN54LS273	8-Bit Latch	MIL-883, Class B
1	SN54LS90	Counter	MIL-883, Class B
5	SN5417	Hex Buffer	38510, Class B
10	CD4049	Hex Inverter	MIL-883, Class B
5	CD4050	Hex Buffer	MIL-883, Class B

Table F-1. Parts List - Digital Subsystem (Cont)

Quantity	Part Number	Description	Procurement Level
8	54C906	Hex Buffer	MIL-883, Class B
1	AM686	High Speed Comparator	MIL-883, Class B
8	HA2730	Dual Op-Amp	MIL-883, Class B
4	HA2520	High Performance Op-Amp	MIL-883, Class B
2	HA245	Line Transmitter	MIL-883, Class B
2	DG191	Analog Switch	MIL-883, Class B
1	DAC-08A	8-Bit D/A Converter	MIL-883, Class B
1	DAC-331-10	10-Bit D/A Converter	MIL-883, Class B
4	889-10K	10K Resistor Network	IRS/ER
8	2N5841	NPN Transistor	IRS/TXV
4	2N4260	PNP Transistor	IRS/TXV
16	1N5711	Diode	IRS/TXV
4	1N4099	Zener Diode, 6.8V	JTXV
1	1N752A	Zener Diode, 5.6V	JTXV
30	RCR	Resistor, Carbon Comp	MIL ER
45	RNC	Resistor, Metal Film	MIL ER
45	M39014/01	Capacitor, Ceramic	MIL ER
15	CMR	Capacitor, Mica	MIL ER

' Motorola Custom LSI Circuit

Table F-2. Parts List - RF Section

Quantity	Part Number	Description	Procurement Level
119	CKR	Capacitor, Ceramic	MIL ER
5	CSR	Capacitor, Tant	MIL ER
8	CLR	Capacitor, Tant Foil	MIL ER
132	CDR	Capacitor, Cer Chip	MIL ER
2	CWR	Capacitor, Tant Chip	MIL ER
13	—	Variable Capacitor	—

Table F-2. Parts List - RF Section (Cont)

Quantity	Part Number	Description	Procurement Level
134	RCR	Resistor, Carbon Comp	MIL ER
85	RNC	Resistor, Metal Film	MIL ER
56	WA4XXXJ	Resistor, Chip	—
22	RTH	Termistor	—
4	—	Resistor Network	—
2	1N751A	Diode	JTXV
1	1N829	Diode	JTXV
2	1N4148	Diode	JTXV
2	1N5148	Diode	JTXV
2	1N5140	Diode	JTXV
4	1N5711	Diode	JTXV
1	HP5082-0112	SRD	—
2	HP5082-0300	SRD	—
1	HP5082-6459	Diode	—
1	2N2222A	Transistor	JTXV
1	2N2857	Transistor	JTXV
1	2N2907A	Transistor	JTXV
2	2N3251A	Transistor	JTXV
5	2N3421	Transistor	JTXV
8	NE02107	Transistor	—
2	NE64480	Transistor	—
1	KJ6001	Transistor	—
2	HA2520	High Performance Op-Amp	MIL-883, Class B
1	MC4344F	Phase Detector	MIL-883, Class B
2	MC10137L	Decade Counter	MIL-883, Class B
10	MIC76T	RF Amp	MIL-883, Class A
2	MIC236	RF Mixer	MIL-883, Class A
2	MIC336	RF Mixer	MIL-883, Class A

Table F-2. Parts List - RF Section (Cont)

Quantity	Part Number	Description	Procurement Level
12	MS750873	Inductor	—
104	—	Inductor, Variable	—
4	—	Inductor	—
1	—	Transformer	—
5	—	SW Quartz Crystal	—
20	—	Feed Thru	—